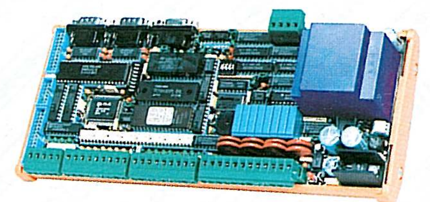

elektor
Nº 262
3,61 €
REVISTA INTERNACIONAL DE ELECTRONICA Y ORDENADORES
INTERFAZ SMS PARA GSM
control remoto de procesos mediante teléfono GSM
Transceptor de infrarrojos para PCs
Interfaz de disco duro para puerto de impresora
Curso básico de Microcontrolador
Interrogador maestro
Kit de reconocimiento de voz
SIEMENS
09.11.01
15:37
Menu
MEDER electronic DIP05-1A72-12D
C25/S25/350B
LARPRESS, S.A.
8 414090 101455
00262

# Para el control y la automatización industrial: amplia posibilidad de elegir entre centenares de tarjetas profesionales abaco®

## ZBR xxx   ZBT xxx

**Versión con relé**
**Versión con transistor**

Esta familia de placas periféricas, para el montaje en barras DIN, comprende: doble sección de alimentación; una para la lógica de bordo y para la CPU exterior y la otra para la sección aislada galvánicamente; 4 modelos con un número distinto de entradas optoaisladas y salidas con Relé. Se encuentran también las versiones equivalentes ZBT xxx con salidas con Transistores. Configuraciones de Entrada + Salidas disponibles: ZBR 324=32x24; ZBR 246=24+16; ZBR 168=16+8; ZBR 84=84+4. Se manejan por medio de **Abaco® I/O BUS.** Son el complemento ideal para las CPU de la **Serie 3** y **Serie 4** con las cuales se combinan mecánicamente en la misma barra DIN formando un dispositivo único y sólido. Se pueden manejar directamente por medio del adaptador **PCC-A26**, desde la puerta paralela del PC.



## GPC® 15R

No necesita de ningún sistema de extensión externo. 84C15 con cristal de cuarzo de 20MHz, Z80 compatible. Hay amplia disponibilidad de lenguajes de programación como por ejemplo: FGDOS, NSB8, PASCAL, C, FORTH, BASIC Compiler, etc. Puede controlar directamente Display LCD y teclado. Doble alimentador incorporado y contenedor para barra de Omega. Hasta 512K RAM con batería de Litio y 51 2K Flash; Real-Time Clock; 24 líneas de E/S TTL; 8 Relé; 16 entradas optoacopladas; 4 Contadores optoacoplados; Buzzer; 2 líneas serie en RS 232, RS 422, RS 485 o Current Loop; conector para expansión Abaco® I/O BUS; Watch Dog; etc. A través del sistema operativo FGDOS administra discos RAM y ROM y programa directamente la FLASH incorporada con el programa del usuario.

## QTP 24
### Quick Terminal Panel 24 Teclas

Panel operativo, IP 65, de bajo coste con 4 tipos de Display distintos. Está disponible con display LCD Retroiluminado o Fluorescente en los siguientes formatos: 2 x 20 o 4 x 20 caracteres; 16 LED, Buzzer, carpetas de personalización, E² con capacidad de más de 200 mensajes; comunicación serie en RS232, RS422, RS485 o Current Loop; alimentador incorporado, etc. Opción para lectores de Fichas Magnéticas; Relé de consentimiento Real-Time Clock con batería de Litio. Es muy fácil de utilizar en cualquier tipo de ambiente.

## GPC® 323D

No hace falta ningún sistema de extensión externo. Dallas 80C320 de 22MHz o de 30MHz. Están disponibles muchos lenguajes de programación, como por ejemplo: PASCAL, C, BASIC etc. Es capaz de pilotar directamente Display LCD y teclado. Alimentador incorporado y contenedor para barra Omega. No se necesita ningún sistema de desarrollo y con el FMO52 puede programar la FLASH con el programa del usuario; 32K RAM; zócalo para 32K EPROM, zócalo para 32K RAM, EPROM o EEPROM; 24 líneas de E/S; 11 líneas de A/D de 12 Bits; Contador y Timer; E² serie; Reloj con batería de Litio; Buzzer; 2 líneas serie RS 232, RS 422, RS 485 o Current Loop; Watch-Dog; Conector de expansión por Abaco® I/O BUS.

## GPC® AM4

Placa de la **Serie 4** de 5 x 10 cm con **CPU Atmel ATmega 103** de 552MHz con 128K FLASH; 4K RAM y 4K EEPROM internos más 32K RAM externos; 16 líneas de I/O; Temporizador / Contador; 3 PWM; 8 A/D de 10 bit; RTC con batería al litio; 1 serial en RS232; RS422; RS485 o bien Current Loop; Watch Dog; Conector para Abaco® I/O BUS; montaje en **Piggy-Back**, programación de la FLASH en ISP compatible **Equinox**; etc. Herramientas software como **BASCOM, Assembler, Compilador, C, BASIC**, etc.

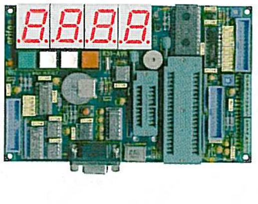
## UEP 48

3 años de garantía

Programador Universal de alta velocidad con zócalo de 48 pins. No necesita adaptadores para dispositivos DIL tipo EPROM, EEPROM serie, FLASH, EEPROM paralela, GAL, µP, etc. Se suministra con software de control, alimentador externo y cable paralelo para conexión al PC.

## K51 AVR

La placa **K51-AVR** permite efectuar una completa experimentación tanto de los distintos dispositivos pilotables a **I²C-BUS** como las posibilidades ofrecidas por las **CPU** de la fam. **8051** y **AVR** sobre todo en combinación con el rellenador **BASCOM**. Numerosos ejemplos y data-sheet disponibles en nuestro sitio.



## MP PIK   MP AVR-51

**Programador** de **Bajo Coste**, para **µP PIC** o bien para **MCS51** y **Atmel AVR**. Además puede programar las **EEPROM** seriales en IIC, **Microwire** y **SPI**. Se suministra con software y alimentador de red.



## ER 05

Borrador para UV EPROM de bajo coste capaz de borrar hasta dispositivos de 32 pins. Timer incluido. Se suministra con alimentador externo.

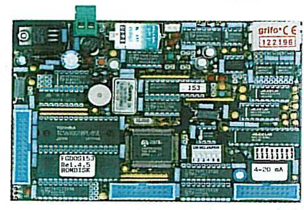
## E P 3 2

3 años de garantía

Es un Programador Universal **económico** para EPROM, FLASH, E² serie, EEPROM. Utilizando los oportunos adaptadores opcionales programa también GAL, microprocesadores, E² serie, etc. Está dotado de software, alimentador externo y cable para el puerto paralelo del PC.

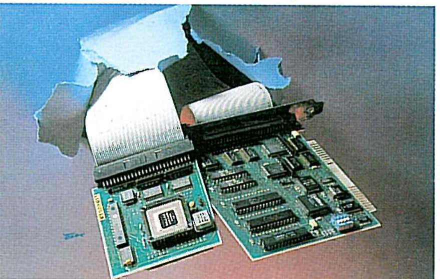
## GPC® 11

68HC11A1 de 8MHz; 32K RAM; 32K EPROM,: 8K RAM + Reloj con batería de Litio; 0,5K E²; 8 líneas de conversores A/D; 32 líneas de E/S; línea serie RS 232, RS 422 o RS 485; Watch-Dog; Contador y Timer; etc. Bajo consumo: 0,25W. Alimentador incorporado y contenedor para barra Omega. El complemento ideal para el Compilador ICC-11 o Micro-C.



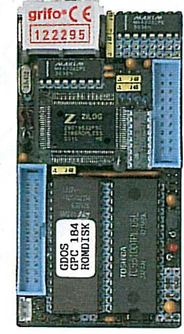
## GPC® 153

No hace falta ningún sistema de extensión externo. 84C15 de 10MHz Z80 compatible. Están disponibles muchos lenguajes de programación, como por ejemplo: FGDOS,NSB8, PASCAL, C, FORTH, BASIC etc. Es capaz de pilotar directamente Display LCD y teclado. Alimentador incorporado y contenedor para barra Omega. 512K RAM con batería de Litio; 512K FLASH; 24 líneas de E/S; 8 líneas de conversores A/D de 12 Bits; Contador y Timer; E² serie; Real-Time Clock con batería de Litio; Buzzer; 2 líneas serie RS 232, RS 422, RS 485 o Current Loop; Watch-Dog; Conector de expansión por Abaco® I/O BUS; etc. Programa directamente la FLASH con el programa del usuario.



## ICEmu-51/UNI

Es un potente emulador **In-Circuit** Profesional en **tiempo real**, de tipo universal para la familia de microprocesadores 51 hasta 42 MHz de emulación. Amplia variedad de Pod, para los diversos microprocesadores, a partir de los 51 genéricos; Dallas, Siemens; Philips; Intel; Oki; Atmel; etc. Trace memory; Breakpoints; Debugger de elevado nivel; etc.



## GPC® 184

**General Purpose Controller Z8S195**

Placa de la **Serie 4** de 5 x 10 cm. Con CPU Z8S195 con cuarzo de 22 MHz, código computable Z80; de hasta 512 K RAM; de hasta 512K FLASH con gestión de RAM-ROM DISCO; RTC con batería al Litio; 16 I/O; conector de batería al Litio exterior; 2 lineas seriales; una RS 232 más una RS232, RS422, RS485 o Current-Loop; Watch-Dog; Temporizador; Contador, etc. Programa directamente la FLASH de bordo por medio del **OS FGDOS** ofrecido en promoción **GRATUITAMENTE** en esta placa. Conector de expansión para **Abaco® I/O BUS**; montaje en **Piggy-Back**. Muchas herramientas software como **PASCAL, C, BASIC**, etc.




**QBM - Quality By Measurement** - C/MONTENEGRE, 15-17, ANTLO. 2ª - 08029 BARCELONA
Tel. 93 490 10 28 / 93 490 10 61 - E-mail: ventas@qbm.es - **Web sites:** http://www.qbm.es

**40016 S. Giorgio di P. (BO) Italy - Via dell'Artigiano, 8/6- Tel. +39 051 892.052 (r.a.) - Fax +39 051 893.661**
**E-mail: grifo@grifo.it - Web sites: http://www.grifo.it - http://www.grifo.com**







Director
Eduardo Corral

Colaboradores
Jose Mª Villoch, Pablo de la Muñoza, Andrés Ferrer.

Redacción
VIDELEC, S.L.
Santa Leonor, 61 4º-1
28037 MADRID
Tel.: 91 375 02 70
Fax: 91 375 61 42

Publicidad
*Director de Publicidad:* Julio Mollejo
julio.mollejo@larpress.com
*Coordinadora de publicidad:* Gema Sustaeta
C/ Medea, 4 5ª planta (Edificio ECU)
Tel.: 91 754 32 88
Fax: 91 754 18 58
28037 MADRID
email: publicidad@lar.es

*Delegación Cataluña*




*Delegado:* Isidro Ángel Iglesias López
iai_ad_press@infonegocio.com
*Jefe de publicidad:* Eva Matute Calvo
emc_ad_press@infonegocio.com
Comte d'Urgell, 165-167, B-1º-3ª
08036 BARCELONA
Tel.: 93 451 89 07
Fax: 93 451 83 23
email: ad_press@infonegocio.com

Suscripciones
C/ La Forja, 27
28850 Torrejón de Ardoz (Madrid).
Tels.: 91 677 70 75 - Fax: 91 676 76 65

Edita




Director Editor
Julio Rodríguez

Director de Producción
Gregorio Goñi
Director Comercial
Eloy Zamanillo

Distribución en España
COEDIS, S.A.
Ctra. Nacional II Km. 602,5
08750 Molins de Rei - Barcelona
Tel.: 93 680 03 60

Importador exclusivo Cono Sur:
CEDE, S.A.
Distribuidor en capital: Huesca y Sanabria
Distribuidor en interior: D.G.P
Importador para Chile:
Iberoamericana de Ediciones, S.A.
C/ Leonor de la Corte, 6035. Quinta Normal
Santiago de Chile
Tel.: 774 82 87 - 774 82 88
Distribución en Chile:
Alfa, S.A.
Distribución México:
Importador exclusivo Cade, S.A. de C.V.
C/ Lago Ladoga, 216
Colonia Anahuac-Delegación
Manuel Hidalgo. 11320 Mexico D.F.
Tel: 5254-2999 Fax: 5545-6879
Distribución Estados: Citem
Distribución D.F.: Unión de Voceadores
Distribución en Venezuela:
Distribuidora Continental
Distribución en Colombia:
Disunidas, S.A.
Distribución en Ecuador:
Disandes
P.V.P. en Canarias, Ceuta y Melilla 600 Ptas.

Imprime
Gráficas Reunidas C/ Mar Tirreno, 7 Bis. Polígono Industrial San Fernando. 28830 San Fernando de Henares. Madrid.
Depósito legal: GU.3-1980
ISSN 0211-397X
31/Marzo/2.002

Preimpresión
Videlec, S.L.
C/ Sta. Leonor, 61 -4º local 1

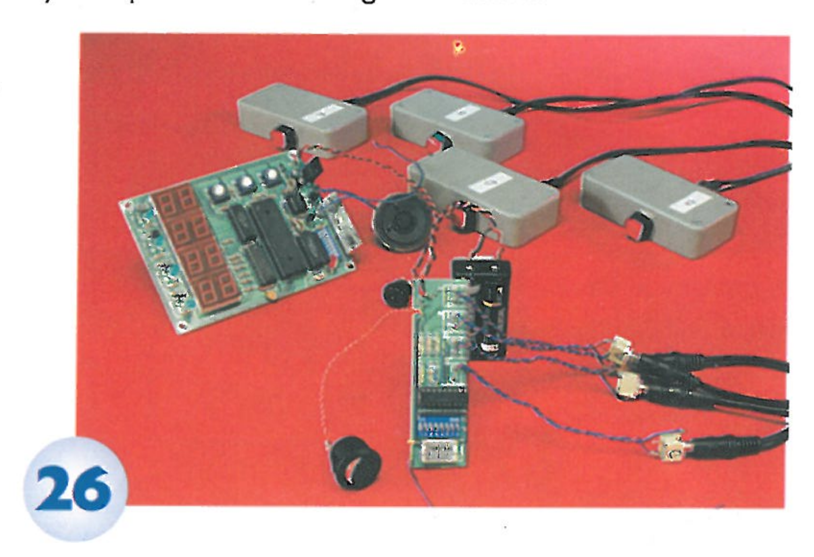







## Interrogador Maestro

El interrogador maestro que presentamos en este montaje es una versión de cuatro jugadores con un jugador extra que controla la "caja" y que realiza las preguntas. No se necesita cableado entre el jugador y el dispositivo del interrogador maestro.




## Control Remoto de Procesos Mediante un Teléfono Móvil

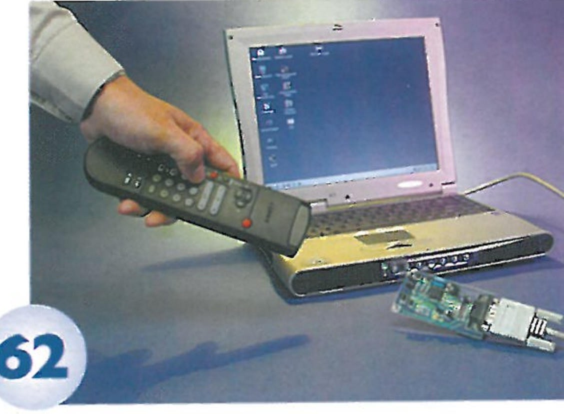
El Sistema de Mensajes Cortos (SMS) de un teléfono móvil estándar tiene muchas más aplicaciones que las de intercambiar mensajes escritos. El montaje de esta aplicación nos permite hacer trabajar al teléfono móvil en una localización remota, verificando y controlando equipos externos.



# CONTENIDO

## Transceptor de Infrarrojos para PC (I)

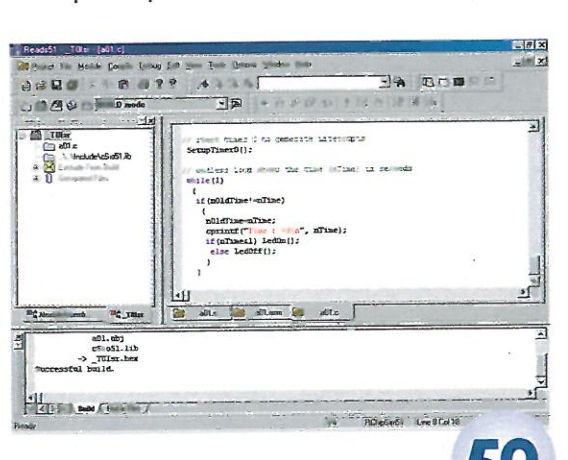
Este artículo se encarga tan sólo de describirnos un transceptor de infrarrojo (IR) que podemos utilizar con nuestro PC para control remoto y transferencia de datos. Basado en la exitosa serie de artículos "Diseño de Periféricos Serie para PC" de Elektor, también nos proporciona información sobre los principios de transferencia de datos por infrarrojos y de la tecnología del interfaz, que podemos utilizar para nuestros propios proyectos.

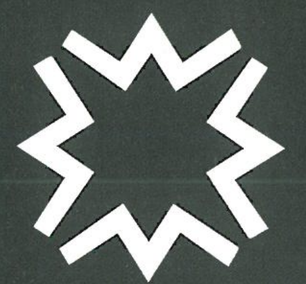



# ARTÍCULOS INFORMATIVOS

## Curso Básico de Microcontrolador (I)

Este curso está indicado para todo aquel que desee saber cómo trabajan los microcontroladores y cómo utilizarlos, pero que nunca se ha atrevido a preguntarlo. Está pensado para explicar los conocimientos fundamentales, empezando desde el inicio. La placa microcontrolador Flash 89S8252 de Elektor sirve como plataforma hardware.





# MATELEC

## Salón Internacional de Material Eléctrico y Electrónico

## International Exhibition of Electrical and Electronic Equipment

# 2002

8-12 • OCTUBRE/OCTOBER

Parque Ferial Juan Carlos I - Madrid




Transportista Oficial
Official Transporter

Parque Ferial Juan Carlos I • 28042 Madrid
Apdo. de Correos 67 067 • 28080 Madrid • España/Spain
Tel.: (34) 91 722 50 34 / 50 00 • Fax: (34) 91 722 57 91
e-mail: matelec@ifema.es • www.matelec.ifema.es



IFEMA
Feria de Madrid

# USB UART (2)

## parte 2: configuración de los puertos corrientes

**Aunque la interface USB publicada anteriormente en Elektor permite configurar la corriente del puerto sólo para un puerto, el circuito integrado del puerto permite configurar la corriente en cada pin.**

En el módulo Uart USB tenemos la subrutina Sub WrIsink Pin, Wert. El parámetro Pin puede tomar un valor de 0 a 7 para los pines de los puertos P00 a P07 o de 8 a 11 para los pines de los puertos P10 a P13. El listado 1 muestra un simple programa para la configuración de la corriente para las cuatro salidas del puerto Port 1, mientras que la Figura 1 muestra la pantalla asociada.

El microcontrolador contiene un simple DAC de 4 bit para cada pin del puerto, que consta de cuatro pesos corrientes. Nosotros podemos preguntarnos si podemos usar esto para algo más que para controlar el brillo de unos pocos LEDs. En principio, lo único que necesitamos es convertir la corriente controlada en una tensión de salida. Una resistencia de 200 W conectada al puerto 1 (ver Figura 2) nos proporciona un buen rango de tensión de salida para una corriente consumida de hasta 15 mA.

Para chequear la linealidad de este dispositivo, la tensión de salida se medirá para las 16 configuraciones de corriente. El resultado se muestra en la Tabla 1. Esos gráficos se han evaluado de forma gráfica utilizando Excel. Como se puede ver en la Figura 3, la linealidad es buena.

La corriente de salida se puede utilizar para construir una fuente de alimentación simple programable. La Figura 4 muestra un sencillo circuito utilizando un amplificador operacional de potencia L272. El ajuste de cero y cambio de flanco se puede realizar independientemente. Si es necesario, el rango de ajuste se puede hacer más pequeño, por ejemplo de 3,5 a 5 V.

### Un sencillo conversor A/D

También podemos utilizar la configuración del D/A de los pines del puerto para construir un sencillo conversor A/D. En este caso, todo lo que tenemos que hacer es medir valores de resistencia. La resistencia a medir se debe conectar entre un puerto del pin y Vcc. El método está basado en la configuración de la corriente de entrada necesaria para tener en el pin del puerto la tensión a nivel bajo, dando de esta manera el valor de resistencia conectada.

La Figura 5 nos muestra cómo puede obtenerse una simple función de entrada analógica conectando ocho potenciómetros al puerto 1. Para el puerto 0, nos dará mejor resultado un ajuste de entre 2 a 12 KΩ. Para el puerto 1 necesitaremos valores menores de resistencia, porque este puerto tiene una corriente de entrada mayor.

Para cada conversión analógica, la corriente de entrada se incrementa en un paso hasta que el comparador cambie de estado. Esto significa que cada medida requiere 16 transacciones de salida y 16 transacciones de entrada. Debido a que un control que accede a través del puerto USB requiere 4 ms, cada medida tarda 256 ms. Por lo tanto, un completo set de medidas para los ocho canales tardará unos 2 s.

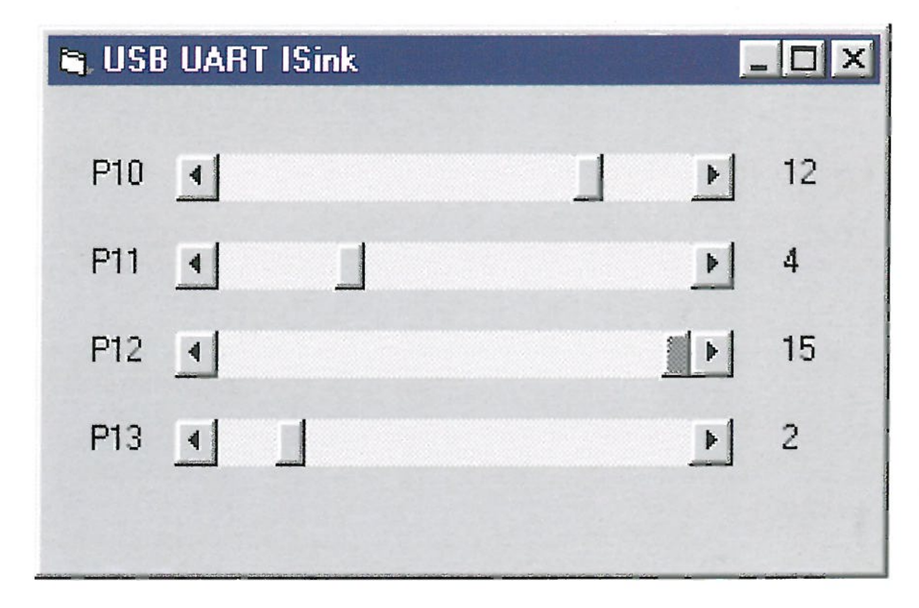



Figura 1. Pantalla de visualización para el programa LED.FRM.

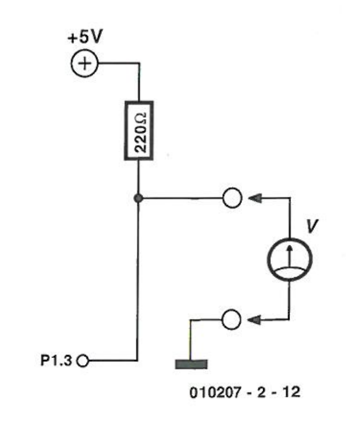



Figura 2. Control de una tensión de salida a través de P1.3.

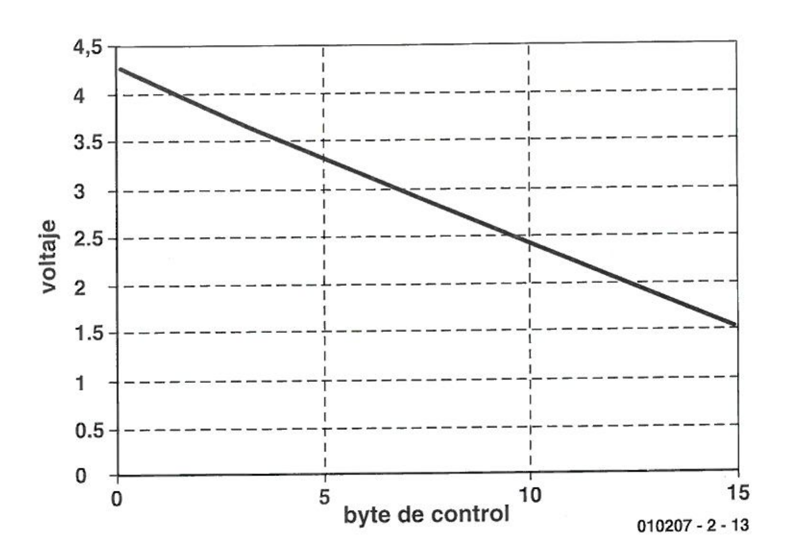



Figura 3. Tensión de salida vs. Valor del bytes de control.

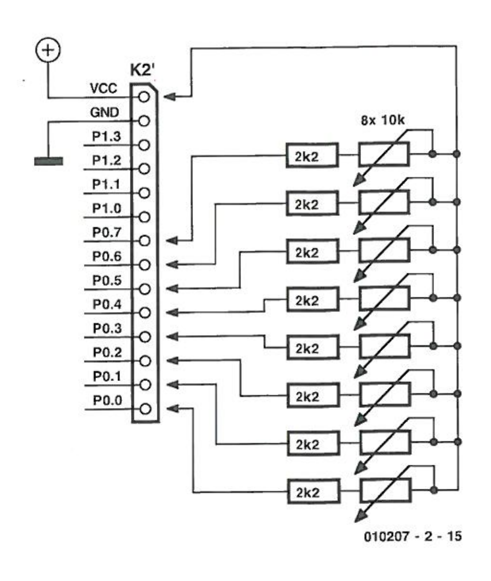



Figura 5. Conexión de potenciómetros para entradas analógicas.

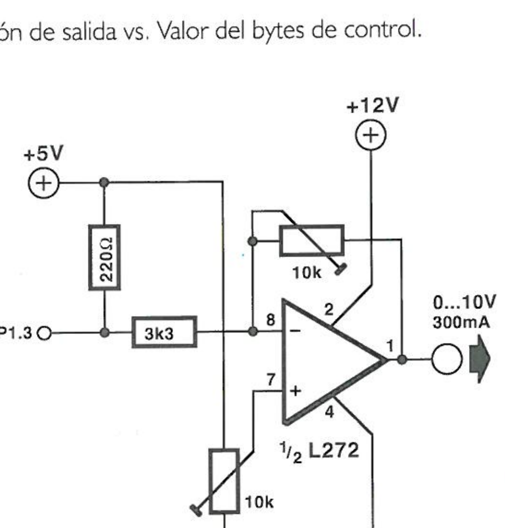
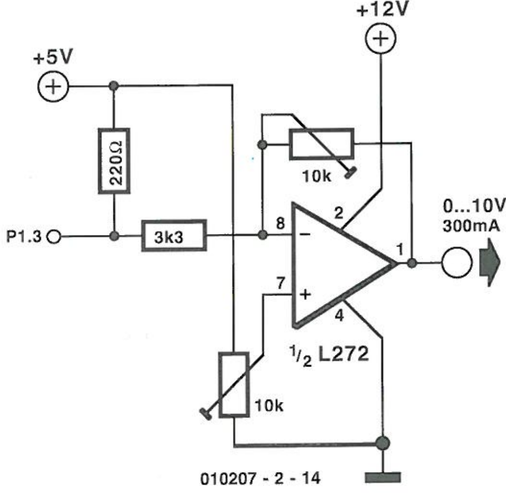




Figura 4. Una fuente de tensión programable.

USB UART AD 4 Bit

| | | |
|---|---|---|
| P00 | | 13 |
| P01 | | 12 |
| P02 | | 11 |
| P03 | | 10 |
| P04 | | 9 |
| P05 | | 11 |
| P06 | | 13 |
| P07 | | 14 |

Figura 6. Display analógico de 8 canales.

**Tabla 1.**
**Tensiones medidas en P1.3.**

| Saltos de corriente | Tensión |
|---|---|
| 0 | 4,29 V |
| 1 | 4,09 V |
| 2 | 3,90 V |
| 3 | 3,70 V |
| 4 | 3,51 V |
| 5 | 3,32 V |
| 6 | 3,15 V |
| 7 | 2,96 V |
| 8 | 2,79 V |
| 9 | 2,61 V |
| 10 | 2,43 V |
| 11 | 2,24 V |
| 12 | 2,06 V |
| 13 | 1,89 V |
| 14 | 1,71 V |
| 15 | 1,54 V |

**Listado 1.**
**USBuart2.vbp**

```
Private Sub Form_Load()
  WrPort0 Wert
End Sub

Private Sub HScroll1_Change()
  Wert = HScroll1.Value
  WrIsink 8, Wert
  Label5.Caption = Str$(Wert)
End Sub

Private Sub HScroll2_Change()
  Wert = HScroll2.Value
  WrIsink 9, Wert
  Label6.Caption = Str$(Wert)
End Sub

Private Sub HScroll3_Change()
  Wert = HScroll3.Value
  WrIsink 10, Wert
  Label7.Caption = Str$(Wert)
End Sub

Private Sub HScroll4_Change()
  Wert = HScroll4.Value
  WrIsink 11, Wert
  Label8.Caption = Str$(Wert)
End Sub
```

En el listado 2 se muestra un programa para chequear y visualizar las configuraciones de los ocho puertos. Aquí usaremos desplazadores como indicadores en lugar de elementos de control. Los desplazadores mostrados en la pantalla se controlan de forma remota por medio de potenciómetros

(010207-2)

**Listado 2.**
**Utilización de la UART-USB como conversor A/D de 8 canales**

```
Function ADCh0()
  WrIsink 0, 0
  Ain = 0
  While ((RdPort0 And 1) > 0) And (Ain < 15)
    Ain = Ain + 1
    WrIsink 0, Ain
  Wend
  ADCh0 = Ain
End Function

Function ADCh1()
  WrIsink 1, 0
  Ain = 0
  While ((RdPort0 And 2) > 0) And (Ain < 15)
    Ain = Ain + 1
    WrIsink 1, Ain
  Wend
  ADCh1 = Ain
End Function

Function ADCh2()
  WrIsink 2, 0
  Ain = 0
  While ((RdPort0 And 4) > 0) And (Ain < 15)
    Ain = Ain + 1
    WrIsink 2, Ain
  Wend
  ADCh2 = Ain
End Function

Function ADCh3()
  WrIsink 3, 0
  Ain = 0
  While ((RdPort0 And 8) > 0) And (Ain < 15)
    Ain = Ain + 1
    WrIsink 3, Ain
  Wend
  ADCh3 = Ain
End Function

Function ADCh4()
  WrIsink 4, 0
  Ain = 0
  While ((RdPort0 And 16) > 0) And (Ain < 15)
    Ain = Ain + 1
    WrIsink 4, Ain
  Wend
  ADCh4 = Ain
End Function

Function ADCh5()
  WrIsink 5, 0
  Ain = 0
  While ((RdPort0 And 32) > 0) And (Ain < 15)
    Ain = Ain + 1
    WrIsink 5, Ain
  Wend
  ADCh5 = Ain
End Function

Function ADCh6()
  WrIsink 6, 0
  Ain = 0
  While ((RdPort0 And 64) > 0) And (Ain < 15)
    Ain = Ain + 1
    WrIsink 6, Ain
  Wend
  ADCh6 = Ain
End Function

Function ADCh7()
  WrIsink 7, 0
  Ain = 0
  While ((RdPort0 And 128) > 0) And (Ain < 15)
    Ain = Ain + 1
    WrIsink 7, Ain
  Wend
  ADCh7 = Ain
End Function

Private Sub Form_Load()
  WrPort0 Wert
End Sub

Private Sub Timer1_Timer()
  Value = ADCh0()
  HScroll1.Value = Value
  Label9.Caption = Str$(Value)
  Value = ADCh1()
  HScroll2.Value = Value
  Label10.Caption = Str$(Value)
  Value = ADCh2()
  HScroll3.Value = Value
  Label11.Caption = Str$(Value)
  Value = ADCh3()
  HScroll4.Value = Value
  Label12.Caption = Str$(Value)
  Value = ADCh4()
  HScroll5.Value = Value
  Label13.Caption = Str$(Value)
  Value = ADCh5()
  HScroll6.Value = Value
  Label14.Caption = Str$(Value)
  Value = ADCh6()
  HScroll7.Value = Value
  Label15.Caption = Str$(Value)
  Value = ADCh7()
  HScroll8.Value = Value
  Label16.Caption = Str$(Value)
End Sub
```




# CEBEK®

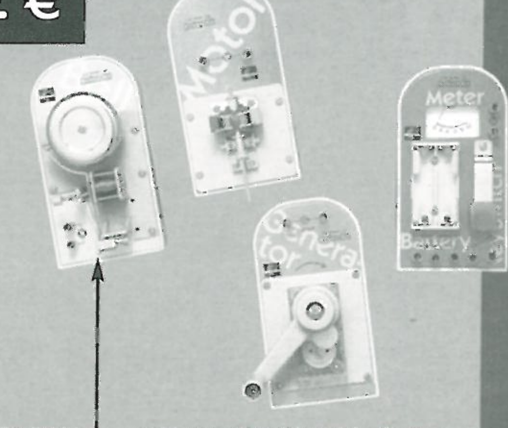



ELECTRONICS CIRCUITS

## ¿APRENDER TECNOLOGIA ? ¡ES FACIL Y DIVERTIDO!

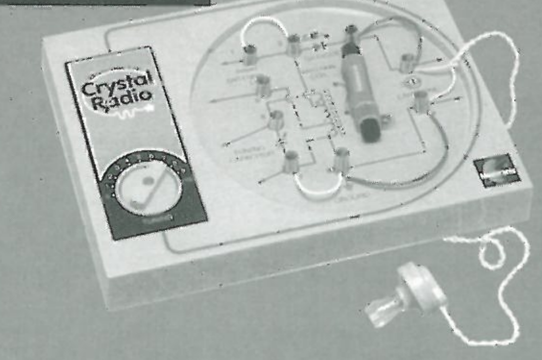
ENSEÑANZA TECNICA Y RIGUROSA,
PERO ENTRETENIDA Y ATRACTIVA
RESULTADO:SE APRENDE SIN DARSE CUENTA
PARA TODOS LOS NIVELES ECONOMICOS
PARA TODOS LOS NIVELES EDUCATIVOS

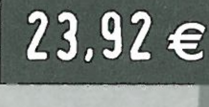
45,62 €




MX - 901 LA RADIO DE CRISTAL

11,12 €



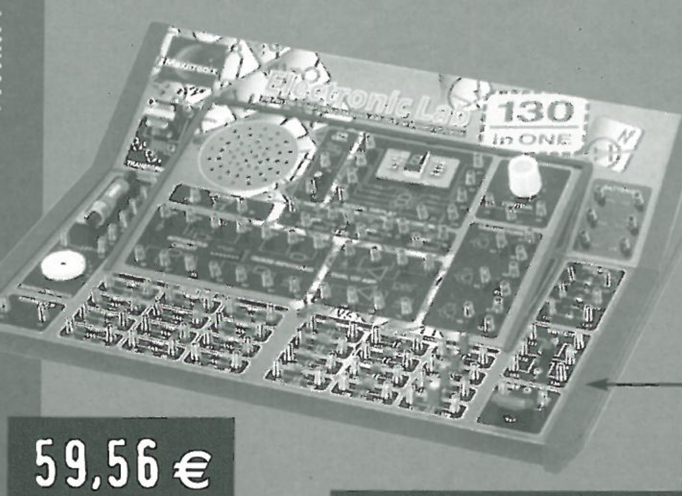



MX - 902 MINI LABORATORIO ELECTRICO 4 EN 1

I.V.A. NO INCKUIDO EN LOS PRECIOS
P.V.R.: PRECIO DE VENTA RECOMENDADO

23,92 €




MX - 903 ENTRENADOR 30 EN 1

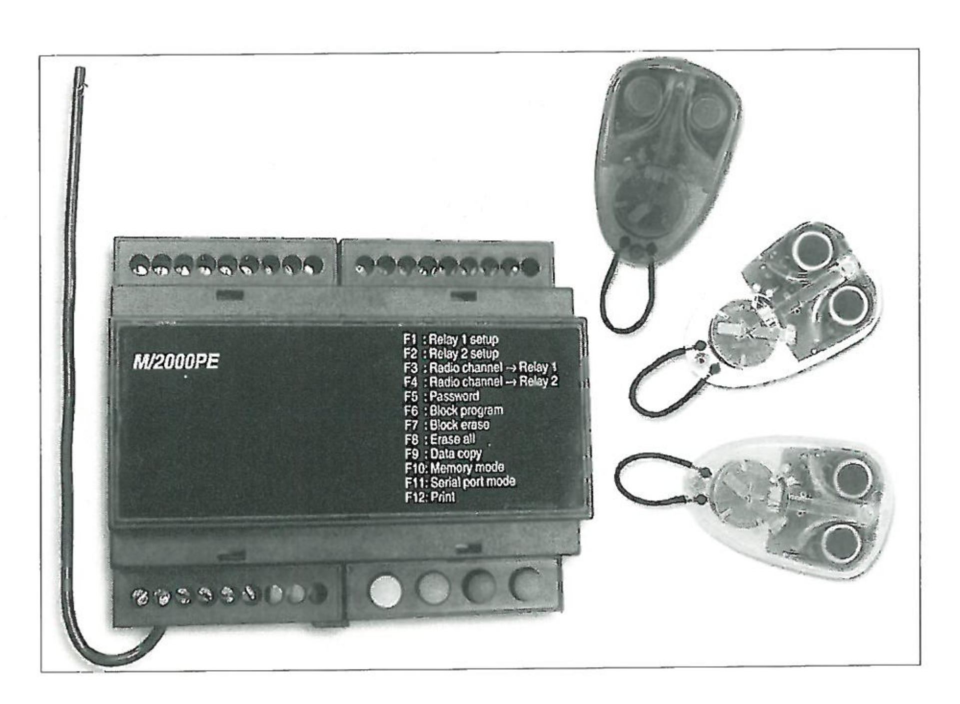
59,56 €

MX - 906 ENTRENADOR 130 PRACTICAS



CEBEK.COM

## Sistema de Apertura de Puertas Via Radio




Circontrol presenta un completo sistema de apertura de puertas vía radio. Este equipo permite el control de un acceso a una distancia de 100 metros.

Los mandos a distancia disponen de un modo de encriptación de los datos que hacen que en ningún caso se pueda captar la señal desde una antena externa para después volverla a emitir.

El controlador del sistema permite controlar hasta 2 accesos, los cuales pueden estar vinculados a la existencia de un lazo magnético de detección de vehículos en el caso de la instalación en un parking.

Al mismo tiempo dispone de la posibilidad de asociar un pulsador al relé para poder accionar la puerta desde el interior sin necesidad de utilizar el mando a distancia.

Como otras opciones cabe destacar la posibilidad de implementar control antipassback y la de dar de alta códigos en bloque.

*Para más información pueden dirigirse al Departamento Comercial de CIRCONTROL de la calle Baldrich, 222 en 08223-TERRASSA. Tel. 93.736.29.40 y Fax 93.736.29.41 E-mail: circontrol@circutor.es Web: www.circontrol.com*

## Memorias SRAMS de bajo consumo y alta velocidad de 8 Mbits para productos industriales y de consumo

Hitachi anuncia una nueva serie de memorias SRAM de bajo consumo de 8 Mbits diseñadas para terminales de información portátiles industriales y de bajo voltaje, como los productos de comunicación, control y medición. Las HM62V16512, HM62V16514 y HM62V8100 ofrecen operación a alta velocidad y se encapsulan en CSP (Chip Scale Package) de 48 patillas de 6.5 m m x 9.8 m m que permite a los diseñadores crear productos más pequeños y más ligeros.

También está disponible un encapsulado estándar TSOP-II (Thin Small Outline Package) de 44 patillas para la compatibilidad con la SRAM de bajo consumo de 4 Mbits de Hitachi.

Utilizando un proceso CMOS de 0.18 µm y técnicas de diseño de circuitos especiales, la nueva serie ofrece operación a alta velocidad y bajo voltaje. La corriente operativa de 30 mA (máximo) a 55 ns y la corriente de retención de datos de 25 µA (máximo) permite también asegurar la permanencia de los datos durante largos periodos de tiempo en backup de batería en dispositivos portátiles. Además de proporcionar un tiempo de acceso de lectura/escritura de datos de alta velocidad y la buena retención de datos citada, las SRAMs de bajo consumo y alta densidad permiten simplificar el diseño del sistema ya que las operaciones de refresco no son necesarias.

Junto con su tiempo de acceso rápido (55 ns) y amplia gama de temperatura de operación (-40°C a +85°C), esta serie incluye diferentes combinaciones de grado de corriente (versión L de bajo consumo o versión SL de consumo súper bajo) y encapsulados (TSOP-II 44 o CSP 48) para satisfacer una gran variedad de necesidades desde aplicaciones industriales hasta aplicaciones de consumo portátiles.

Ya están disponibles muestras y la producción masiva empezará de inmediato.

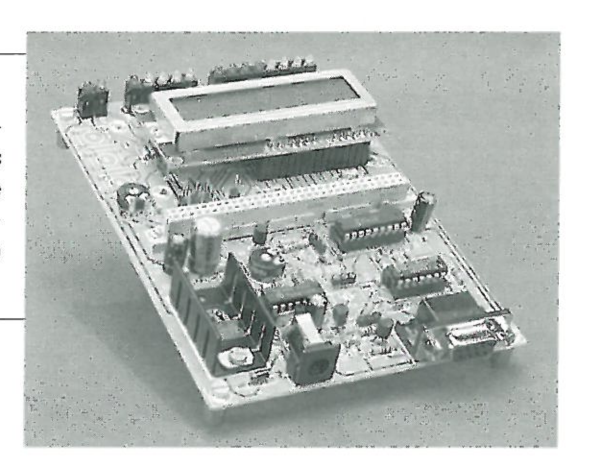
*Para más información sobre los productos y servicios ofrecidos por el Grupo de Componentes Electrónicos de Hitachi Europe, visite la página Web en: http://www.hitachi-eu.com/semiconductors/ o envíe un email a web.eca@hitachi-eu.com*

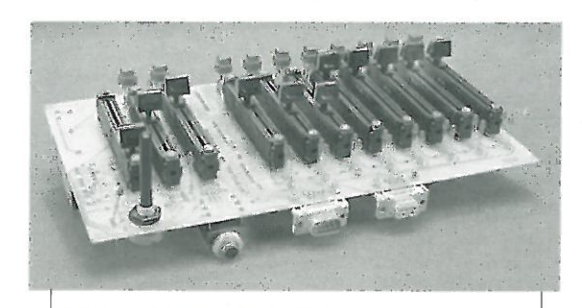
**Especificaciones**

| Nombre | Organización | Encapsulado | Velocidad | Corriente | Gama de Temp. |
|---|---|---|---|---|---|
| HM62V16514L TTI | 512k x 16 bits | TSOP 44 | 55ns | L, SL | -40°C a +85°C |
| HM62V16512LBPI | 512k x 16 bits | CSP 48 | 55ns | L, SL | -40°C a +85°C |
| HM62V81 OOL TTI | 1 M x 8 bits | TSOP 44 | 55ns | L, SL | -40"C a +85°C |
| HM62V8100LBPI | 1M x 8 bits | CSP 48 | 55ns | L, SL | -40°C a +85°C |

## Sistema de desarrollo PIC 16F84

Este ordenador de placa simple que contiene un 'PICee-E' es un sistema de desarrollo educativo para el ya popular microcontrolador PIC16F84 de Microchip. Los 35 comandos del controlador se pueden comprobar sin ningún tipo de hardware adicional, y todas las aplicaciones que hemos tenido en cuenta, desde un simple LED que parpadea a un reloj, se pueden implementar utilizando este bonito diseño.



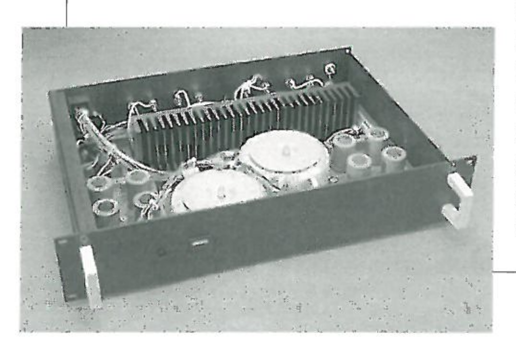


## Etapa multicontrol para iluminación de discoteca

Esto no es sino una mesa de mezclas analógica que se utilizará en primer lugar como etapa de iluminación y en pequeñas discotecas. El diseño se puede ampliar a cualquier número de canales de luz desde una tensión de control estándar de 0 a 10 V. Además, está disponible el Master, Control y Flash.

## Amplificador de audio de 100 W

El diseño de este amplificador está basado en el circuito integrado amplificador de potencia AF tipo TDA 7293 de ST Microelectronics. Cuatro de estos están disponibles para conexión individual, como si se tratase de una configuración puente utilizando dos pares, haciéndolos ideales para altavoces activos o modos biamplificadores.



## También...

Circuitos integrados multipropósito para constructores de maquetas, comprobadores manuales de S/PDIF, amplificador de instrumentación, EEDTs Pro 1.2, control DMX para relés...



MARZO 2002

## CONDICIONES GENERALES

Los circuitos impresos, carátulas autoadhesivas, ROMs, PALs, GALs, microcontroladores y disquetes que aparecen en las páginas de ELEKTOR se encuentran a disposición de los lectores que lo requieran. Para solicitarlos es necesario utilizar el cupón de pedido que se encuentra en las páginas anexas.
Este mismo cupón también puede utilizarse para efectuar pedidos de los libros de la colección de ELEKTOR (en versión original inglesa).
- Los items marcados con un asterisco (*) tienen una vigencia limitada y su disponibilidad solo puede garantizarse durante un cierto periodo de tiempo.
- Los items que no se encuentran en esta lista no están disponibles.
- Los diseños de circuitos impresos se encuentran en las páginas centrales de la Revista. En ocasiones y por limitación de espacio no se garantiza la publicación de todos los circuitos. En estos casos los lectores interesados pueden solicitar los diseños, utilizando el mismo cupón de pedido y les serán enviados a su domicilio contra reembolso de 500 pts. (incluidos gastos de envio).
- Los EPROMs, GALs, PALs, (E)PLDs, PICs y otros microcontroladores se suministrarán ya programados.
Los precios y las descripciones de los diferentes productos estan sujetos a cambios. La editorial se reserva el derecho de modificar los precios sin necesidad de notificación previa. Los precios y las descripciones incluidas en la presente edición anulan los publicados en los anteriores números de la Revista.

## FORMA DE ENVIO

Los pedidos serán enviados por correo a la dirección indicada en el cupón de las páginas anexas. Además los lectores pueden formular pedidos por teléfono llamando al número 91 327 37 97 de lunes a viernes en horario de 9,30 a 14 h y de 16 a 19 h. Fuera de este horario existe un contestador telefónico preparado para recoger las demandas. Los gastos de envio serán abonados por el comprador, tal como se indica en el cupón.

## FORMA DE PAGO

Todos los pedidos deberán venir acompañados por el pago, que incluirá los gastos de envio, tal como se indicó anteriormente.
El pago puede realizarse mediante cheque conformado de cualquier banco residente en territorio español, giro postal anticipado, tarjeta VISA (en este caso debe indicarse la fecha de caducidad, domicilio del propietario de la tarjeta y firma del mismo).
Nunca se deberá enviar dinero en metálico con el pedido. Los cheques y los giros postales deben ser nominativos a la orden de VIDELEC S.L.

## SUSCRIPCIONES A LA REVISTA Y EJEMPLARES ATRASADOS

Las suscripciones o pedido de números atrasados, si se encuentran disponibles, se realizarán a LARPRESS, C/ Medea, 4 5ª planta (Edificio ECU) 28037 Madrid.
Los precios de ejemplares atrasados son de 600 pts más gastos de envio.

## COMPONENTES UTILIZADOS EN LOS PROYECTOS

Todos los componentes utilizados en los proyectos ofrecidos en las páginas de la Revista se encuentran generalmente disponibles en cualquier establecimiento especializado o a través de los anunciantes de este ejemplar. Si existiera alguna dificultad especial con la obtención de alguna de las partes, se indicará la fuente de suministro en el mismo artículo. Lógicamente los proveedores indicados no son exclusivos y cualquier lector podrá optar por su suministrador habitual.

## CONDICIONES GENERALES DE VENTA

**Plazo de entrega:** El plazo normal será de 2-3 semanas desde la recepción del pedido. No obstante no podemos garantizar el cumplimiento de este periodo para la totalidad de los pedidos.
**Devoluciones:** Aquellos envios que se encuentren defectuosos o con la falta de alguno de los componentes podrán ser devueltos para su reposición, solicitando previamente nuestro consentimiento mediante llamada telefónica al número (91) 3273797 en horario de oficina. En este caso la persona que llame recibirá un número de devolución que deberá hacer constar al devolver el material en un lugar bien visible. En este caso correrá por nuestra cuenta el gasto de envio de la devolución, debiéndolo hacer así constar el remitente en su oficina postal. A continuación se le enviará nuevamente el pedido solicitado sin ningún gasto para el solicitante.
En el caso de que la devolución se realice por otra causas ajenas a la revista, sólo se admitirá si el material devuelto se encuentra en perfectas condiciones para ser vendido de nuevo. En este caso al remitente le será devuelto el importe previamente enviado, reteniendo un 10 % del precio para cubrir los gastos de manipulación y embalaje.
En cualquiera de los casos anteriores, solo se admitirán las devoluciones en un plazo de tiempo de 14 dias contados a partir de la fecha de envio del pedido.
**Patentes:** Algunos de los circuitos o proyectos publicados pueden estar protegidos mediante patente, tanto en la Revista como en los libros técnicos. La editorial LARPRESS no aceptará ninguna responsabilidad derivada de la utilización inadecuada de tales proyectos o circuitos para fines distintos de los meramente personales.

No obstante, los diseños de circuitos impresos si pueden ser utilizados para uso personal y privado, sin necesidad de obtener un permiso previo.
**Limitación de responsabilidad:** Todos los materiales suministrados a los lectores cumplen la Normativa Internacional en cuanto a seguridad de componentes electrónicos y deberán ser utilizados y manipulados según las reglas universalmente aceptadas para este tipo de productos. Por tanto ni la editorial LARPRESS, ni la empresa suministradora de los materiales a los lectores se hacen responsables de ningún daño producido pos la inadecuada manipulación de los materiales enviados.

## CONSULTORIO TECNICO

Existe un Consultorio técnico telefónico gratuito a disposición de todos los lectores. Este sevicio se presta todos los lunes y martes laborables en horario de 17 a 19 h.
**El número de teléfono para consultas es el 91 375 02 70.**

| | Código | Precio (€) |
|---|---|---|
| ***E262 MARZO 2002*** | | |
| **Interfaz de disco duro para puerto de impresora:** | | |
| - PCB | 010047-1 | 00,00 |
| - Disk, project software | 010047-11 | 00,00 |
| - 7064LC84-15, programmed | 010047-31 | 00,00 |
| **Iluminación y caja de cambios:** | | |
| - Disk, project software | 010204-11 | 00,00 |
| - PIC16C57, programmed | 010204-41 | 00,00 |
| **Curso Básico de Microcontrolador:** | | |
| - PCB | 010208-1 | 00,00 |
| - Disk, courseware | 010208-11 | 00,00 |
| **Interrogador maestro:** | | |
| - PCB, transmitter and receiver | 010030-1 | 00,00 |
| - Disk, project software | 010030-11 | 00,00 |
| - PIC17C44-16/P, programmed | 010030-41 | 00,00 |
| ***E261 FEBRERO 2002*** | | |
| **Placa microcontroladora flash para 89S8252:** | | |
| - PCB | 010208-1 | 32,00 |
| - Disk, project software | 010208-11 | 11,00 |
| **Medidor de descarga/capacidad de batería:** | | |
| - PCB set | 010201-1 | 34,03 |
| - Disk set, project sofware | 010201-11 | 19,00 |
| - ST62T65B6, programmed | 010201-41 | 40,00 |
| **Cerradura electrónica codificada:** | | |
| - PCB | 004003-1 | 22,54 |
| - Disk, project sofftware | 006001-1 | 11,00 |
| - PIC16F84-04/P, programmed | 006501-1 | 31,28 |
| **Fuente de alimentación digital para laboratorio:** | | |
| - PCB | 000166-1 | 25,00 |
| - Disk set, project software | 000166-11 | 13,44 |
| - PIC16F84A-04P, programmed 1A version | 000166-41 | 43,00 |
| - PIC16F84A-04P, programmed 2.5 version | 000166-42 | 43,00 |
| **Control remoto RC5:** | | |
| - Disk, project software | 000189-11 | 11,00 |
| - Attiny22L-8PC, programmed | 000189-41 | 20,00 |
| **UART USB:** | | |
| - PCB | 010207-1 | 37,93 |
| - Disk, project software | 010207-11 | 18,00 |
| - CY7C63001A, programmed | 010207-41 | 63,02 |
| - Set: PCB + 010207-11 + 010207-41 | 010207-C | 86,00 |
| ***E260 ENERO 2002*** | | |
| **Control remoto PCM en miniatura (2):** | | |
| - Transmitter PCB | 010205-1 | 23,52 |
| - Receiver PCB | 010205-2 | 19,84 |
| - 87LPC768FN, programmed | 010205-41 | 37,36 |
| - 87LPC762BN, programmed | 010205-42 | 23,20 |
| - Disk, project software | 010205-11 | 11,01 |
| **Medidor de capacidad y descarga de batería:** | | |
| - PCB, includes discharger PCB | 010201-1 | 34,53 |
| - ST62T65, programmed | 010201-41 | 49,16 |
| - Disk, project software | 010201-11 | 19,24 |
| **Demultiplexor DMX de 8 canales:** | | |
| - PCB | 010002-1 | 41,05 |
| - EPROM 27C256 (programmed) | 010002-21 | 18,91 |
| - Disk, project software | 010002-11 | 13,64 |
| ***E259 DICIEMBRE 2001*** | | |
| **Analizador de códigos de IR:** | | |
| - 87LPC764, programmed | 010029-41 | 25,88 |
| - disk, source code | 010029-11 | 11,02 |
| **Saltador:** | | |
| - PCB | 010038-1 | 17,05 |
| - 89C2051, programmed | 010038-41 | 21,33 |
| - disk, source code | 010038-11 | 10,83 |
| **Espionaje de datos en la línea RS232:** | | |
| - PCB | 010041-1 | 10,84 |
| ***E258 NOVIEMBRE 2001*** | | |
| **Programador de Micro AVR Atmel:** | | |
| - PCB | 010005-1 | 28,12 |
| - Disk, project software | 010005-11 | 13,35 |
| - AT89C2051-12PC, programmed | 010005-41 | 21,22 |
| - Set: PCB + 010005-11 + 010005-41 | 010005-C | 52,57 |
| **Módulo gráfico LCD para microprocesadores 8051:** | | |
| - PCB | 000134-1 | 15,23 |
| - Disk, project software | 000134-11 | 10,77 |
| - Set: PCB + 000134-11 | 000134-C | 21,29 |
| **Interface I²C para Servo:** | | |
| - Disk, project software | 010006-11 | 10,77 |
| **Miniservidor WEB personal:** | | |
| - PCB | 010036-1 | 17,93 |
| - Disk, project software | 010036-11 | 10,90 |
| - GAL 16V8, programmed | 010036-31 | 20,90 |
| ***E257 OCTUBRE 2001*** | | |
| **Conversor de 12 a 24V:** | | |
| - PCB | 014025-1 | 20,19 |

# CUPON DE PEDIDO

Por favor envíen este pedido a:
**ADELTRONIK**
**Apartado de Correos 35128**
**28080 Madrid**
**ESPAÑA**
Tel. 91 327 37 97

Nombre
Domicilio

C.P.
Tel. Fax Fecha

**Por favor envíenme los siguientes materiales. Para circuitos impresos, carátulas, EPROMs, PALs, GALs, microcontroladores y disquetes indique el número de código y la descripción.**

| Cant. | Código | Descripción | Precio/unid. IVA incl. | Total € |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Los precios y las descripciones están sujetas a cambio. La editorial se reserva el derecho de cambiar los precios sin notificación previa. Los precios y las descripciones aquí indicadas anulan las de los anteriores números de la revista.

| | |
|---|---|
| Sub-total | |
| Gastos envio | 3,01 |
| Total | |

**Forma de pago (vea la página contigua para más detalles)**
Nota: Los cheques serán en pesetas y conformados por una entidad bancaria.

- ❑ Cheque (nominativo a VIDELKIT, S.L.)
- ❑ Giro postal. Cuenta Postal (ARGENTARIA)
  Nº 1302-9910-37-0022708812
- ❑ VISA Fecha de caducidad: Firma:

Número de tarjeta:

| | Código | Precio (€) |
|---|---|---|
| **Control remoto por infrarrojos para PCs:** | | |
| - AT90S2313, programmed | 000170-41 | 29,02 |
| ***E256 SEPTIEMBRE 2001*** | | |
| **Interface I²C para puerto RS232:** | | |
| - Disk, project sofware | 010045-11 | 10,90 |
| **Dispositivo para concurso:** | | |
| - Disk, project sofware | 000190-11 | 10,96 |
| ***E255 AGOSTO 2001*** | | |
| **PLC DCI:** | | |
| - PCB | 000163-1 | 47,15 |
| - Disk, project software | 000163-11 | 11,62 |
| - Set: PCB + 000163-11 | 000163-C | 54,66 |
| **SMPSU para automóvil:** | | |
| - PCB | 000193-1 | 23,09 |
| **Metrónomo y diapasón:** | | |
| - PCB | 000198-1 | 38,62 |
| - Disk, project software | 000198-11 | 11,55 |
| - PIC 16F84, programmed | 000198-41 | 31,77 |
| **Display de Matriz de Puntos Modular:** | | |
| - Disk, project software | 010021-11 | 11,55 |
| - AT89C2051 programmed | 010021-41 | 22,55 |
| **Tarjeta de 32 canales de entradas analógicas:** | | |
| - PCB | 004090-1 | 29,52 |
| ***E254 JULIO 2001*** | | |
| **Mayor-Domo:** | | |
| - PCB | 000184-1 | 27,60 |
| - Disk, project software | 000184-11 | 11,49 |
| - AT90S8515, programmed | 000184-41 | 59,28 |
| **Control remoto para modelismo:** | | |
| - PCB | 000160-1 | 18,97 |
| - Disk, project software | 000160-11 | 11,49 |
| - COP8782, programmed | 000160-41 | 27,60 |
| **Taladradora para PCB:** | | |
| - PCB | 010024-1 | 52,96 |
| - GAL16R8-25C, programmed | 010024-31 | 16,11 |

| | Código | Precio (€) |
|---|---|---|
| - PIC16C64-20, programmed | 010024-41 | 37,39 |
| - Set: PCB + 010024-31 + 010024-41 | 010024-C | 98,92 |
| **Amplificador de potencia a válvulas HiFi PPP:** | | |
| - PCBs available from Experience Electronics | | |
| ***E253 JUNIO 2001*** | | |
| **Conversor de velocidad de muestreo a 96kHz:** | | |
| - PCB | 010014-1 | 43,62 |
| **Crescendo Edición Millenium:** | | |
| - PCB, amplifier (mono block) | 010001-1 | 26,47 |
| - PCB, power-on delay | 974078-1 | 16,56 |
| **MIDI en el puerto RS232:** | | |
| - PCB | 000139-1 | 31,49 |
| - EPROM 27C256, programmed | 000139-21 | 18,26 |
| - Disk, driver, source code, hex file | 000139-11 | 11,08 |
| - Set: PCB + 000139-21 + 000139-11 | 000139-C | 53,53 |
| ***E252 MAYO 2001*** | | |
| **Luces MIDI y control de diapositivas:** | | |
| - PCB | 000179-1 | 76,76 |
| - EPROM 27C256, programmed | 000179-12 | 38,70 |
| - disk, source code &binary | 000179-11 | 28,38 |
| **ADC 2001 para audio:** | | |
| - PCB, converter | 010017-1 | 39,67 |
| - PCB, power supply | 010017-2 | 21,68 |
| **Generador de pulsos programable:** | | |
| - PCB | 000200-1 | 21,87 |
| - Disk set, project software | 000200-11a/b | 13,54 |
| - PCB + disk set | 000200-C | 32,18 |
| ***E251 ABRIL 2001*** | | |
| **Tarjeta prototipo para Bus PCI (I):** | | |
| - PCB | 010009-1 | 112,95 |
| - disk, Windows software | 010009-11 | 12,69 |
| - GAL22V10, programmed | 010009-31 | 20,94 |
| - disk, DOS software | 010009-12 | 12,69 |
| - PCB, 010009-31 + disk | 010009-C | 146,57 |
| **MCS BASIC-52 V1.3:** | | |
| - Disk, project sofware | 000121-11 | 29,82 |
| - EPROM, programmed | 000121-21 | 39,97 |

| | Código | Precio (€) |
|---|---|---|
| **Controlador de velocidad doble (2):** | | |
| - PCB, SpeedControl + speedPower2 | 000070-4 | 26,65 |
| - PCB, SpeedControl + speedPower1 | 000070-5 | 28 55 |
| - ST62R60BB6, programmed | 000070-41 | 48,23 |
| - Disk, ST6 source code | 000070-11 | 20,94 |
| **Receptor de AM:** | | |
| - PCB | 000176-1 | 34,90 |

### E250 MARZO 2001

| | Código | Precio (€) |
|---|---|---|
| **Decodificadores de control remoto RC5:** | | |
| - PCB | 000081-1 | 17,77 |
| - Disk, project software | 000081-11 | 12,69 |
| - AT90S2343, programmed | 000081-41 | 31,09 |
| **Emulador para la memoria EPROM 27C256 :** | | |
| - PCB | 000153-1 | 46,95 |
| - AT89C2051, programmed | 000153-41 | 24,81 |
| - Disk, project software | 000153-11 | 12,69 |
| - PCB + AT89C2051 + disk | 000153-C | 76,14 |
| **GBPB – Placa de prototipo para Gameboy:** | | |
| - PCB | 000151-1 | 49,5 |
| **Sistema de identificación de llamada vía radio:** | | |
| - PCB, caller unit | 000108-1 | 20,31 |
| - PCB, central receiver | 000108-2 | 20,31 |
| - 3 disk, project software | 000108-11a/b/c | 24,75 |
| - 1 caller PCB + 1 receiver PCB + disk set | 000108-C | 56,47 |
| **Modulador de anchura de pulsos:** | | |
| - Disk, GAL listing | 000123-11 | 12,69 |

### E249 FEBRERO 2001

| | Código | Precio (€) |
|---|---|---|
| **Conversor de sonido a luz PLUS:** | | |
| - PCB | 000107-1 | 51,39 |
| - Project disk | 000107-11 | 12,69 |
| - PIC16F84, programmed | 000107-41 | 31,09 |

### E248 ENERO 2001

| | Código | Precio (€) |
|---|---|---|
| **CAN Adapter for ISA Bus:** | | |
| - PCB | 000071-1 | 64,92 |
| - Project disk | 000071-11 | 13,25 |
| - PCB + project disk | 000071-C | 73,53 |
| **USB Audio-DAC:** | | |
| - PCB | 000169-1 | 23,18 |

### E247 DICIEMBRE 2000

| | Código | Precio (€) |
|---|---|---|
| **e-KEY: Sistema de acceso seguro:** | | |
| - PCB | 000089-1 | 26,38 |
| - disk, source code files | 000089-11 | 17,58 |
| - AT90S1200, programmed | 000089-41 | 28,41 |
| **Cámara sobre Tren de Modelismo:** | | |
| - PCB | 000129-1 | 16,91 |
| **(GBDSO) Osciloscopio de muestreo digital en pantalla de consola Gameboy:** | | |
| - PCB | 990082-1 | 22,32 |
| - disk, DSO Grab and Mathcad demo appl. | 996035-1 | 23,00 |
| - EPROM AT27S256 (PLCC44), programmed | 996528-1 | 37,88 |
| - Set: PCB + 996035-1 + 996528-1 | 990082-C | 74,40 |
| **TV PAL Generador de imagen patrón:** | | |
| - EPM7064, programmed | 990084-31 | 68,32 |
| **Receptor de Onda Corta (OC) Regenerativo:** | | |
| - PCB | 000112-1 | 25,70 |
| **Diseño de periféricos (I):** | | |
| - Set: PCB + 000074-11 | 000074-C | 27,06 |
| - PCB | 000074-1 | 17,59 |
| - Project software | 000074-11 | 13,53 |

### E246 NOVIEMBRE 2000

| | Código | Precio (€) |
|---|---|---|
| **Salida S/PDIF:** | | |
| - PCB | 000131-1 | 26,23 |

### E245 OCTUBRE 2000

| | Código | Precio (€) |
|---|---|---|
| **Modelo digital Märklin para control remoto de trenes:** | | |
| - Set: PCB + 996016-1 | 000066-C | 46,19 |
| - PCB | 000066-1 | 26,77 |
| - Project disk | 996016-1 | 23,43 |
| **Interfaz USB:** | | |
| - Project disk | 000079-11 | 13,39 |
| - PCB | 000079-1 | 14,73 |
| - Set: PCB + 000079-11 + 000079-41 | 000079-C | 48,87 |
| - CY7C63001ACP (programmed) | 000079-41 | 18,74 |

### E244 SEPTIEMBRE 2000

| | Código | Precio (€) |
|---|---|---|
| **Tensión de alimentación simétrica:** | | |
| - PCB | 004064-1 | 11,90 |
| **Lámpara de LED blanco:** | | |
| - PCB | 004024-1 | 8,54 |

### E243 AGOSTO 2000

| | Código | Precio (€) |
|---|---|---|
| **Puerto de I/O de 8 bits:** | | |
| - PCB | 994077-1 | 9,76 |
| **Adaptador para SB Live! Player 1024:** | | |
| - PCB | 004085-1 | 8,89 |
| **Ploteando curvas con HP-GL/2:** | | |
| - Disk, project software | 006005-1 | 10,68 |
| **Implementación del bus I²C:** | | |
| - Disk, project software | 006006-1 | 6,10 |
| - BASIC interpreter in EPROM | 006505-1 | 8,54 |

### E242 JULIO 2000

| | Código | Precio (€) |
|---|---|---|
| **Cerradura inteligente para puertas:** | | |
| - AT89C52-12PC, programmed | 000051-41 | 12,58 |
| - Disk, AT89C52 source code file | 000051-11 | 6,45 |
| - PCB | 000051-1 | 12,79 |
| **Lector de tarjetas magnéticas:** | | |
| - PCB | 000054-1 | 8,06 |
| - AT89C2051-12PC, programmed | 000054-41 | 12,58 |
| - Disk, all project software | 000054-11 | 6,45 |
| - Set: PCB + 000054-11 +000054-41 | 000054-C | 24,18 |
| **Espía de un hilo:** | | |
| - PIC16F84 (programmed) | 000048-41 | 17,74 |
| - PIC16C54 (programmed) | 000048-42 | 14,19 |
| - Disk, all project software | 000048-11 | 6,45 |
| **Interfaz del PC para el Bus CAN:** | | |
| - PCB | 000039-1 | 15,48 |
| - Disk, all project sotware | 006004-1 | 9,73 |

### E241 JUNIO 2000

| | Código | Precio (€) |
|---|---|---|
| **Teclado de funciones especiales:** | | |
| - PCB | 002006-1 | 25,29 |
| - ST62T60(programmed) | 002006-41 | 49,28 |
| - PCB y 002006-41 | 002006-C | 70,03 |
| **Sistema de invención robótico de Lego (2):** | | |
| - PCB | 000040-1 | 12,97 |
| **Medidas mediante Word y Excel:** | | |
| - Disk, Word template and .DLL | 000053-11 | 12,97 |
| **Mezclador MIDI:** | | |
| - PCB | 000021-1 | 24,00 |
| - Disk, AT90S source code files | 996038-1 | 21,40 |
| - 2 x AT90S2313 (a+b), programmed | 996531-1 | 78,46 |
| **Temporizador de reposo RC5:** | | |
| - Disk, PIC source code files | 000026-11 | 12,97 |
| - PIC16F84, programmed | 000026-41 | 31,77 |
| **Pantalla tactil:** | | |
| - Disk, PIC source code & executable | 000055-11 | 12,97 |

### E240 MAYO 2000

| | Código | Precio (€) |
|---|---|---|
| **Estimulador de músculos de bajo impacto:** | | |
| - Disk: source and hex code | 000041-11 | 13,52 |
| - AT89C2051, programmed | 000041-41 | 31,55 |
| - PCB | 000041-1 | 22,45 |
| **Puerto paralelo universal de entrada/salida para PCs:** | | |
| - Set: PCB + 002011-11 | 002011-C | 43,07 |
| - Disk: all project software | 002011-11 | 12,13 |
| - PCB | 002011-1 | 35,18 |

### E239 ABRIL 2000

| | Código | Precio (€) |
|---|---|---|
| **Control de volumen digital:** | | |
| - disk, source code listing | 990080-11 | 11,19 |
| - PCB | 990080-1 | 30,58 |
| - EPROM 27C256 (programmed) | 006506-1 | 16,79 |
| **Receptor de onda media miniatura:** | | |
| - PCB | 000034-1 | 17,99 |
| **Regulador de carga solar:** | | |
| - PCB | 000019-1 | 17,99 |
| **Medidas de temperatura con un DS1621:** | | |
| - Disk, project software | 996027-1 | 11,99 |

### E238 MARZO 2000

| | Código | Precio (€) |
|---|---|---|
| **Convertidor de banda de 2 metros:** | | |
| - PCB | 000013-1 | 23,25 |
| **BASIC 537:** | | |
| - Disk, terminal emulator and sample programs | 996029-1 | 20,86 |
| - EPROM (programmed) | 996532-1 | 25,04 |
| **Control remoto por infrarrojos:** | | |
| - PCB | 990075-1 | 23,85 |
| **Amplificador de audio de potencia en un circuito impreso:** | | |
| - PCB | 000004-1 | 17,88 |
| **Caja de parámetros para programas MIDI:** | | |
| - Set: PCB + 996037-1 + 996521-1 | 990087-C | 56,04 |
| - PIC16F84, programmed | 996521-1 | 29,15 |
| - Disk, PIC source code file and hex-code | 996037-1 | 11,93 |
| - PCB | 990087-1 | 22,66 |

### E237 FEBRERO 2000

| | Código | Precio (€) |
|---|---|---|
| **"Pequeño" ordenador 537:** | | |
| - Disk: 537 Monitor program | 976008-1 | 16,78 |
| - PCB | 990054-1 | 52,86 |
| - GAL 16V8, programmed | 996515-1 | 27,20 |
| - EPROM 27C256, programmed | 976510-1 | 29,51 |
| - set: PCB + 976008-1 + 976510-1 + 996515-1 | 990054-C | 94,91 |
| **DAC 2000 de Audio (3):** | | |
| - GAL 22V10, programmed | 996530-1 | 19,10 |
| - set: PCB + 996530-1 | 990059-C | 65,46 |
| - PCB | 990059-1 | 50,35 |
| **Modelismo ferroviario controlado por PC:** | | |
| - set: PCB + 996522-1 | 990084-C | 51,51 |
| - PIC16F84, programmed | 996522-1 | 41,09 |
| - PCB | 990084-4 | 14,47 |
| **Reloj de la torre Rhine MkII:** | | |
| - PCB | 990076-1 | 59,03 |
| - AT89C2051-12PC, programmed | 996519-1 | 31,83 |
| - Set: PCB + 996519-1 | 990076-C | 86,81 |
| **Placa transformadora:** | | |
| - PCB | 000001-1 | 16,78 |

CEBEK MX=

# CEBEK® SERIE MX

ELECTRONICS CIRCUITS

## MX - 908 ENTRENADOR 300 PRACTICAS

117,50 €

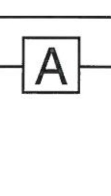
El entrenador 300x1 es uno de los más completos laboratorios de prácticas de electrónica del mercado, que ofrece la posibilidad de realizar 300 prácticas de un nivel medio/alto. A continuación detallamos algunas prácticas:

1 - Transmisor código morse
2 - Contador / divisor por 4
3 - Inducción Electromagnética
4 - Puerta NOR C-MOS
5 - Del binario al BCD
6 - Introducción a la resistencia
7 - Transistor PNP
8 - Transistor NPN
9 - Multivibrador estable
10 - Multivibrador monoestable
11 - Transmisor AM
12 - Contador hexadecimal
13 - Decoder de 7 segmentos BCD
14 - Temporizador digital

Y 286 prácticas más, con los mismos componentes que se suministran en este completo entrenador. Incluye todos el material necesario para poder llevar a cabo las prácticas como: Integrados, resistencias, leds, transistores, cable, diodos, etc...
Se suministra con amplio y detallado manual con las 300 prácticas.
Se recomienda para estudiantes a partir de 16 años.

## MX - 909 ENTRENADOR 500 PRACTICAS

El entrenador 500x1 es el más completo laboratorio de prácticas de electrónica del mercado, que ofrece la posibilidad de realizar 500 prácticas de un nivel alto/superior. Su acabado en formato maletín, facilita poder trasladarlo de una aula a otra.
En el recuadro "A" detallamos algunas de las prácticas. Existen 286 prácticas más hasta hacer un total de 500, con los mismos componentes que se suministran en este completo entrenador. Incluye todo el material necesario para poder llevar a cabo las prácticas como: integrados, placa board, displays LCD, resistencias, leds, transistores, cable, diodos, etc...
Se suministra con amplio y detallado manual con las 500 prácticas.
Se recomienda para estudiantes a partir de 16 años.




1 - Proyectos de software: 100 experimentos
2 - Amplificador operacional CI: 26 experimentos
3 - Circuitos de Radio: 5 experimentos
4 - Circuitos de test y medida: 9 experimentos
5 - Sobre el oscilador: 14 experimentos
6 - Viaje al mundo digital: 25 experimentos
7 - El mundo de los juegos: 12 experimentos
8 - Más sobre circuitos digitales: 14 experimentos
9 - Circuitos que cuentan: 26 experimentos
10 - Electrónica analógica y digital: 9 experimentos

445,71 €

I.V.A. NO INCKUIDO EN LOS PRECIOS - P.V.R.: PRECIO DE VENTA RECOMENDADO



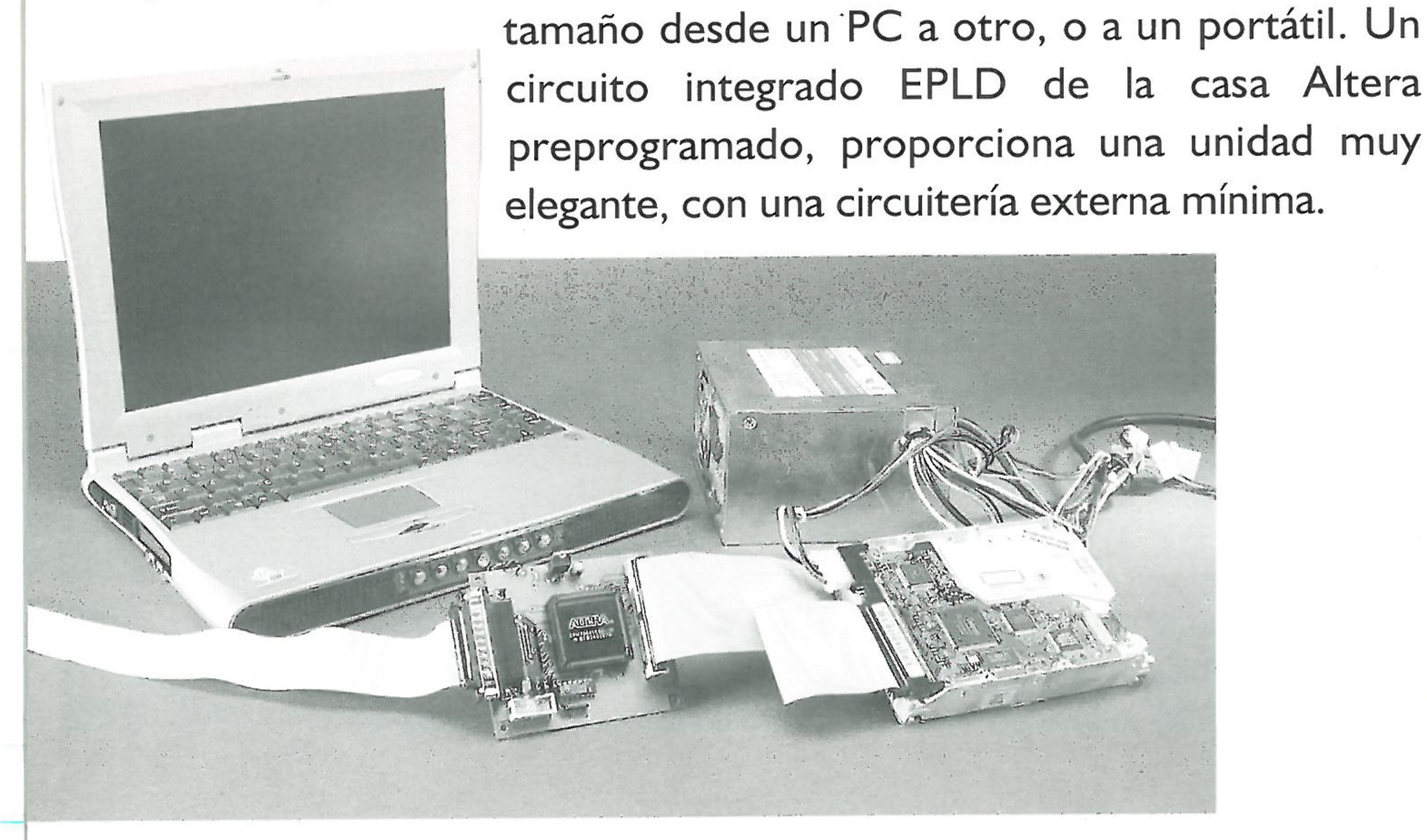


# Interfaz de disco duro para puerto de impresora

## IDE2LPT: almacenamiento removible a través del puerto paralelo

Diseñado por Andrew Buckin (desde una idea de Leonid Slobodchikov)

¿No consigue deshacerse del viejo disco duro de su PC? ¿ Por qué no lo usa como disco portátil? Esta interfaz adaptadora permite utilizar un disco duro interno estándar, de bajo coste, como una unidad portátil de gran capacidad que nos permitirá almacenar o transferir ficheros de gran tamaño desde un PC a otro, o a un portátil. Un circuito integrado EPLD de la casa Altera preprogramado, proporciona una unidad muy elegante, con una circuitería externa mínima.



Es difícil creer que los primeros sistemas de PC's carecieran de disco duro interno. ¿Cómo se las arreglaban entonces? Sólo cuando el disco duro hizo su aparición el PC fue realmente ventajoso, aunque debemos recordar que estos discos sólo podían almacenar unos pocos megabytes y que su tamaño era semejante, en forma y peso, al de un ladrillo de construcción. Aún así, fue una mejora sustancial que evitaba cargar el sistema operativo MS-DOS desde los disquetes de arranque, cada vez que encendíamos el ordenador.

Los modernos discos duros, comparados con los viejos modelos, han mejorado enormemente, convirtiéndose en unos dispositivos específicos

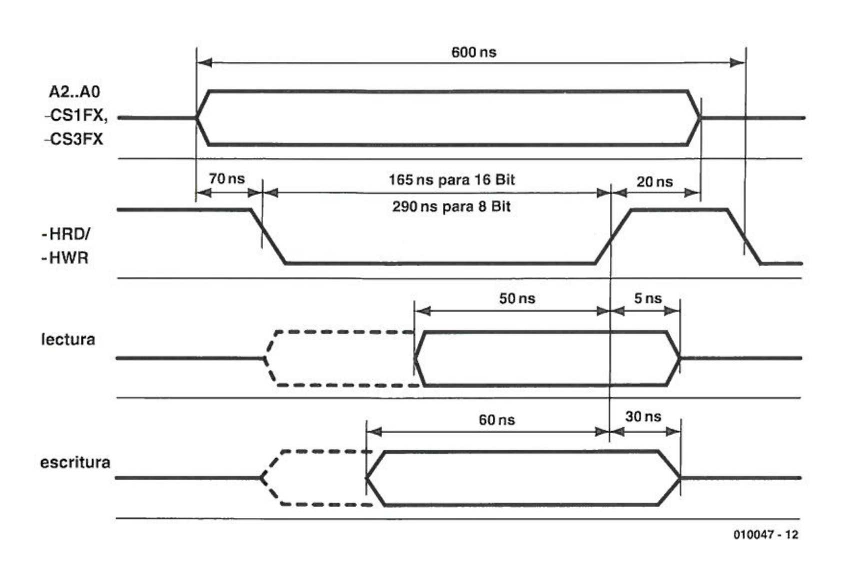



Figura 1. Diagrama de tiempos de lectura y escritura del bus IDE.

## Especificaciones del adaptador:

Máximo tamaño del disco duro: Ilimitado.

Máximo número de discos duros: 1

Velocidad de transferencia: hasta 100 kByte/s

Alimentación: 5 V / < 10 mA (a partir de una fuente de alimentación externa de 12 V)

de una ingeniería de precisión. Los discos flexibles utilizados en aquellos tiempos eran suficiente para almacenar y transferir programas completos, pero hoy día, con el crecimiento progresivo del tamaño de los programas, el uso de un disco duro removible parece la mejor alternativa. Los discos duros internos estándar no removibles ofrecen una alta capacidad de almacenamiento a un bajo coste, pero presentan el inconveniente de que deben conectarse directamente en el interfaz IDE (Integrated Disk Electronic, es decir, Electrónica de Disco Integrada) de la placa base. El PC no dispone de un conector IDE externo como configuración estándar. Si deseamos utilizar un disco duro de este tipo como unidad portátil que se conecte en cualquier PC, será necesario adaptar este tipo de interfaz a un puerto externo del PC ya existente.

Cualquier configuración de PC dispone siempre de un puerto paralelo para impresora, de manera que se puede considerar una buena idea desarrollar una interfaz adaptadora (implementado como un dispositivo CPLD de la casa Altera) que, con algunos programas adicionales, permitiera la transferencia de datos desde y hacia el disco duro por medio del puerto de impresora.

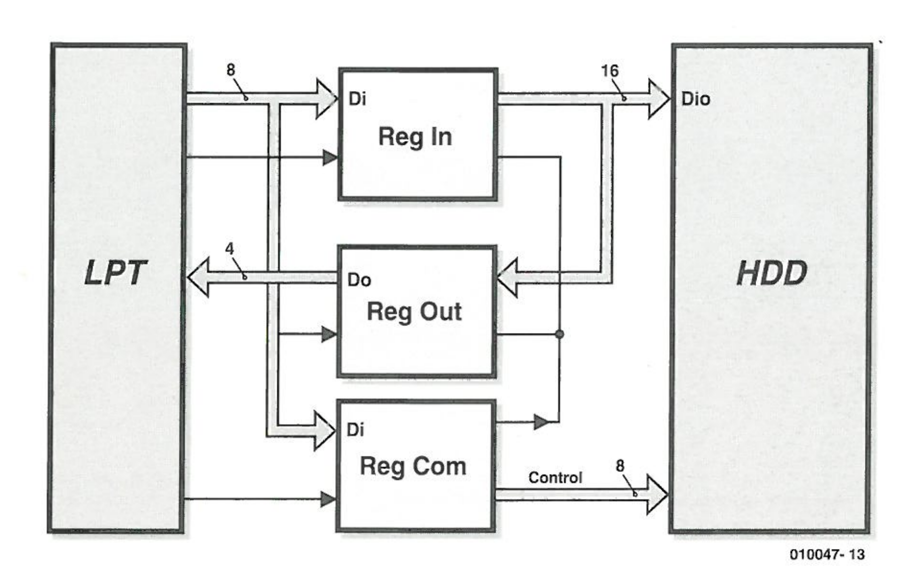



Figura 2. Diagrama de bloques del adaptador LPT/IDE.

**Tabla 1. Distribución de terminales de la interfaz ATA**

| Nº de Terminal | Nombre | Descripción |
|---|---|---|
| 1 | -HRESET | RESET |
| 2 | GND | GND |
| 3 | HD7 | Bit 7 del bus de datos |
| 4 | HD8 | Bit 8 del bus de datos |
| 5 | HD6 | Bit 6 del bus de datos |
| 6 | HD9 | Bit 9 del bus de datos |
| 7 | HD5 | Bit 5 del bus de datos |
| 8 | HD10 | Bit 10 del bus de datos |
| 9 | HD4 | Bit 4 del bus de datos |
| 10 | HD11 | Bit 11 del bus de datos |
| 11 | HD3 | Bit 3 del bus de datos |
| 12 | HD12 | Bit 12 del bus de datos |
| 13 | HD2 | Bit 2 del bus de datos |
| 14 | HD13 | Bit 13 del bus de datos |
| 15 | HD1 | Bit 1 del bus de datos |
| 16 | HD14 | Bit 14 del bus de datos |
| 17 | HD0 | Bit 0 del bus de datos |
| 18 | HD15 | Bit 15 del bus de datos |
| 19 | GND | GND |
| 20 | N/C | Terminal llave |
| 21 | DMARQ | Solicitud de DMA |
| 22 | GND | GND |
| 23 | -HWR | ESCRITURA DE E/S |
| 24 | GND | GND |
| 25 | -HRD | LECTURA DE E/S |
| 26 | GND | GND |
| 27 | IORDY | CANAL DE E/S LISTO |
| 28 | SPSYNC:CSEL | SPINDLE SYNC o CABLE SELECT |
| 29 | -DMACK | DMA ACKNOWLEDGE |
| 30 | GND | GND |
| 31 | INTRQ | Solicitud de interrupción |
| 32 | -IOCS16 | E/S de 16 bits |
| 33 | HA1 | BIT 1 DEL BUS DE DIRCCIONES |
| 34 | -PDIAG | DIAGNÓSTICO PASADO |
| 35 | HA0 | BIT 0 DEL BUS DE DIRCCIONES |
| 36 | HA2 | BIT 2 DEL BUS DE DIRCCIONES |
| 37 | -CS1FX | SELECCIÓN DE CIRCUITO A 0 |
| 38 | -CS3FX | SELECCIÓN DE CIRCUITO A 1 |
| 39 | -DASP | DISCO ACTIVO/DISCO PRESENTE |
| 40 | GND | GND |

## Adaptador LPT a IDE

Anteriormente se publicó un artículo, donde se describía un adaptador que permitía que los controladores de ocho bits pudieran utilizar un disco duro con una interfaz IDE. Aquel artículo describía en detalle la interfaz IDE (también conocida como la interfaz ATA), así como su historia, de manera que no repetiremos en esta ocasión toda esta información. En este caso tan sólo echaremos una ojeada a algunas de las señales más importantes que fueron utilizadas en ese diseño (en la página www.t13.org podemos encontrar una especificación completa de esta interfaz). En la Tabla 1 se muestra una descripción general de las señales presentes en el conector IDE de 40 terminales.

El interfaz LPT2IDE hace uso de las siguientes señales:

**HD15 a HD9:** líneas de datos (ancho del bus de 8 ó 16 bits, bidireccional).

**Tabla 2. Señales y registros: Interfaz LPT y conversor LPT/IDE**

| Direcciones LPTI/LPT2 | Descripción | | | | | |
|---|---|---|---|---|---|---|
| 378h/278h | Registros de datos DATA7... DATA0 (solo escritura) | | | | | |
| | Registro de estado (solo lectura) | | | | | |
| | Bit | Nombre LPT | Nombre IDE2LPT | Estado | Estado de lectura | Función |
| 379h/279h | 7 | OCUPADO | LI3 | I | 0 | DATA 3 desde el conversor |
| | 6 | SLCT | LI2 | I | I | DATA 2 desde el conversor |
| | 5 | PE | LII | I | I | DATA I desde el conversor |
| | 4 | ACK | LI0 | I | I | DATA 0 desde el conversor |
| | 3 | ERROR | — | I | I | — |
| | 2 | — | — | — | 0 | — |
| | I | — | — | — | 0 | — |
| | 0 | — | — | — | 0 | — |
| | Registro de control | | | | | |
| | Bit | Nombre LPT | Nombre IDE2LPT | Estado | Estado de lectura | Función |
| 37ah/27ah | 7 | — | — | — | 0 | |
| | 6 | — | — | — | 0 | |
| | 5 | — | — | — | 0 | |
| | 4 | — | — | — | 0 | |
| | 3 | SLCT_IN | RCWR | 0 | I | Señal de escritura para Reg_COM |
| | 2 | INIT | RLWR | 0 | 0 | Señal de escritura para Reg_IN, desde DATA 7 - DATA 0 a HD7 – HD0 |
| | I | AUTO_FD | RHWR | 0 | I | Señal de escritura para Reg_IN, desde DATA 7 - DATA 0 aHD15 – HD8 |
| | 0 | STROBE | HRESET | 0 | I | Reset HDD |

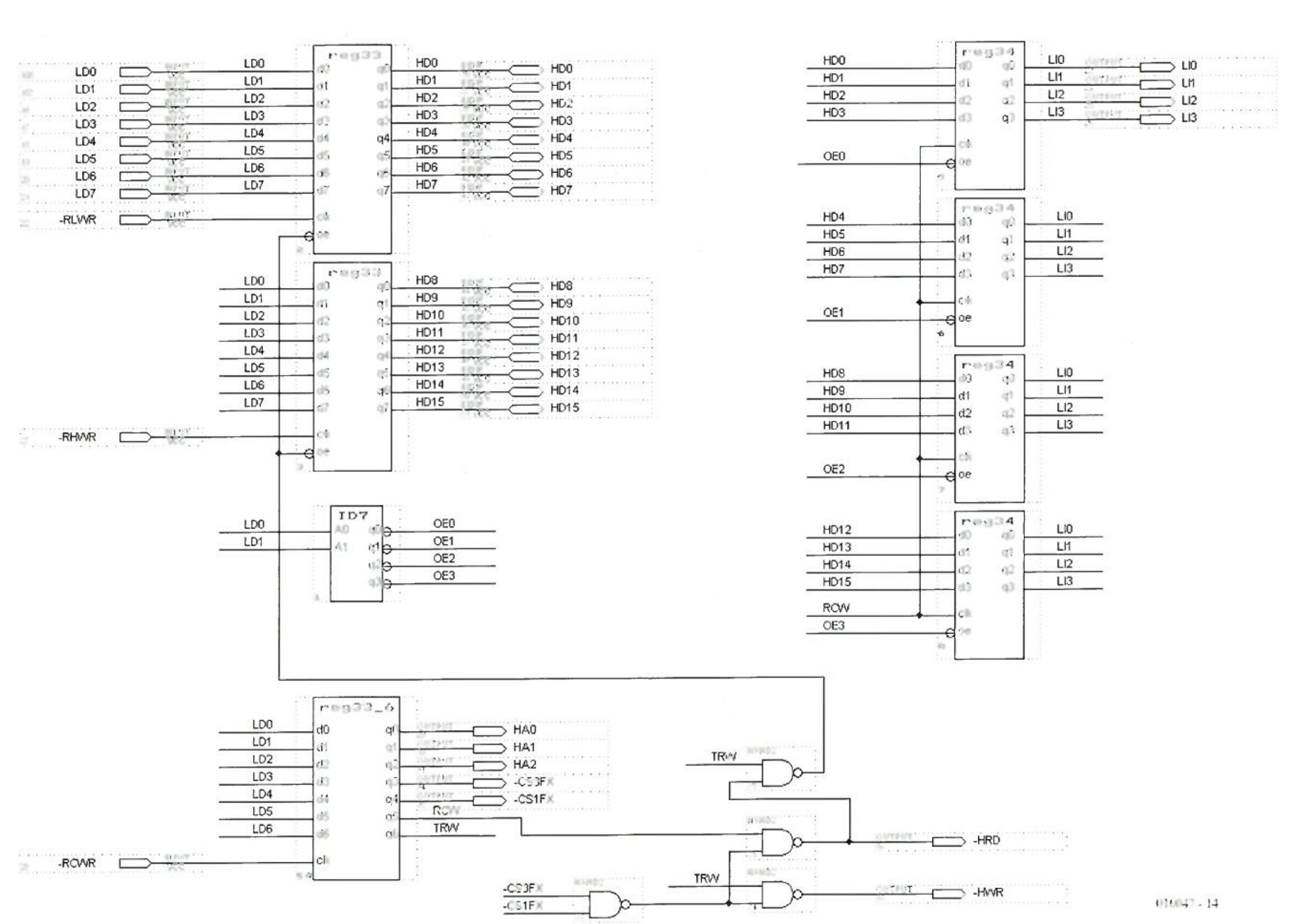



Figura 3. Circuito eléctrico del circuito integrado definido en AHDL y programado en una CPLD.

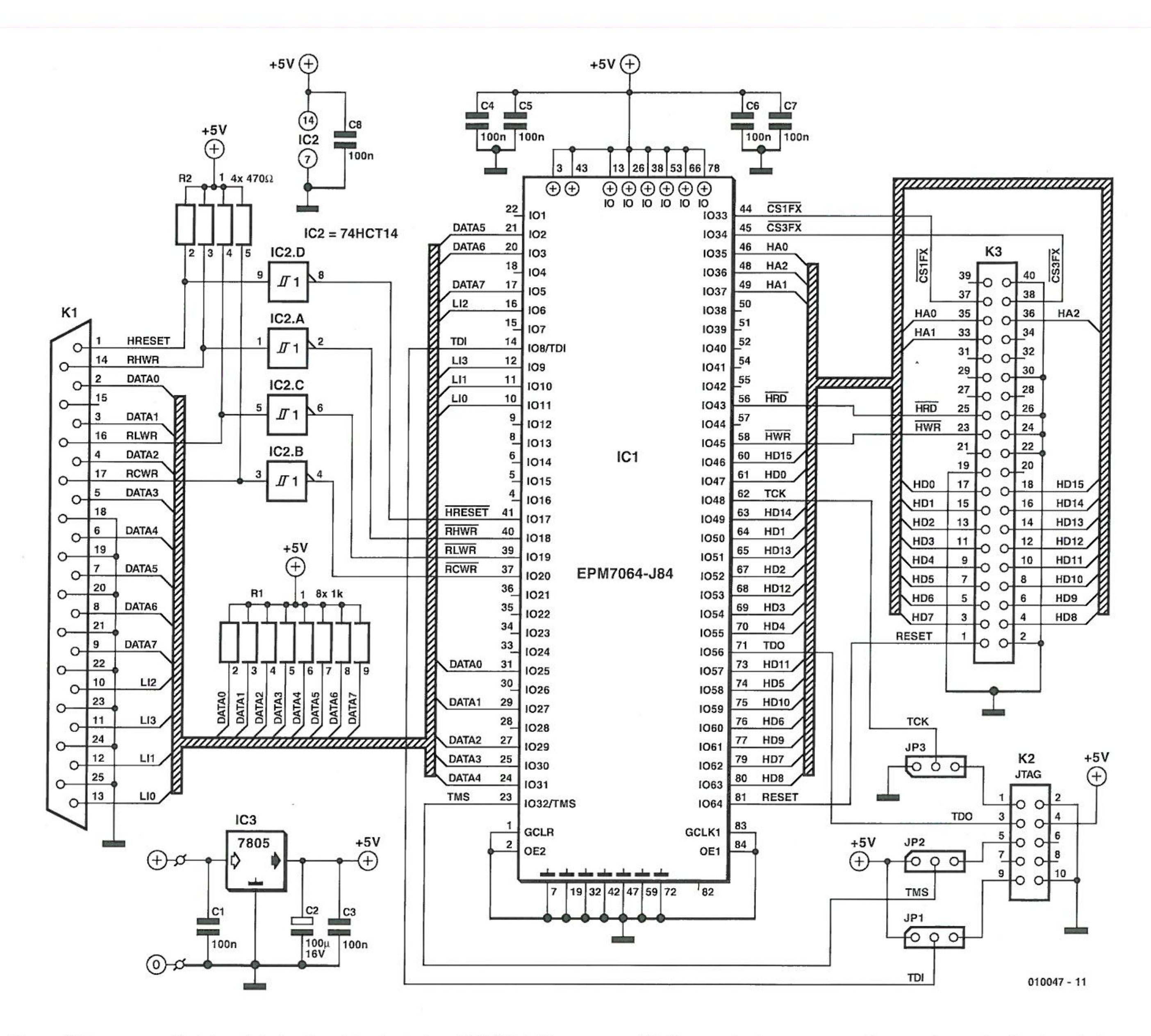



Figura 4.Esquema eléctrico del circuito del adaptador LPT/IDE. El conector JTAG permite la programación en el propio circuito de la CPLD.

**Tabla 3. Señales del registro de control en el conversor LPT/IDE**

| Bit | Nombre | Función | |
|---|---|---|---|
| 7 | — | — | |
| 6 | TRW | Señal de escritura –HWR en el bus IDE. | Activa a nivel 1 |
| 5 | RCW | Señal de lectura –HRD en el bus IDE. | Activa a nivel 1 |
| 4 | -CS1FX | SELECCIÓN DE CIRCUITO 0. | Activo a nivel 0. |
| 3 | -CS3FX | SELECCIÓN DE CIRCUITO 1. | Activo a nivel 0. |
| 2..0 | HA2 - HA0 | BUS DE DIRECCIONES | |

**CS1FX/Chip Select 0:** esta señal selecciona el bloque de registros de comandos.

**CS3Fx(Chip Select 1:** esta señal selecciona el bloque de registros de estado.

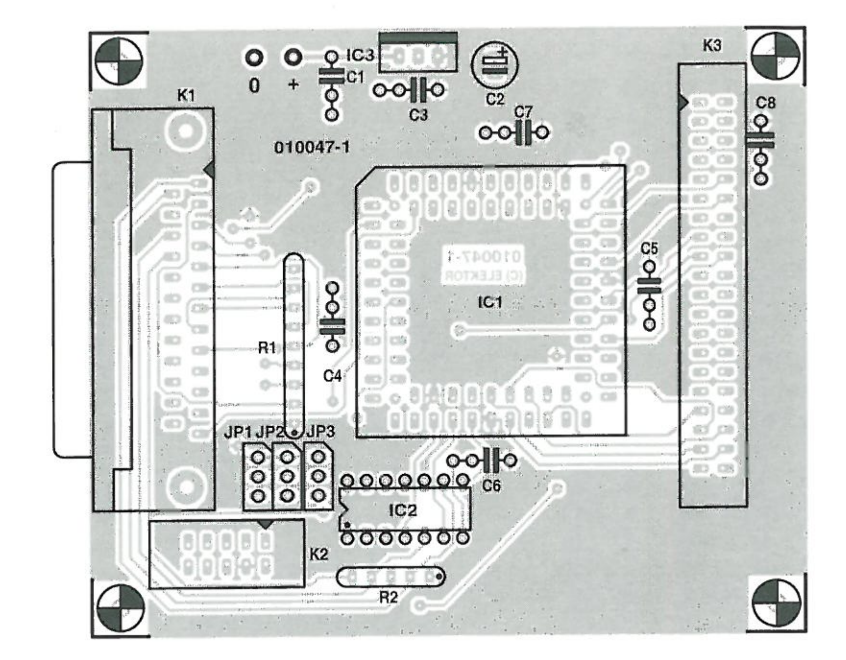
**HRD / I/O read:**señal de lectura para las direcciones del puerto de I/O (E/S). El disco duro colocará el dato en el bus (HD0 a HD7 o HD0 a HD15) cuando estas señales están a nivel bajo y se transfieren al sistema receptor en el flanco de subida.

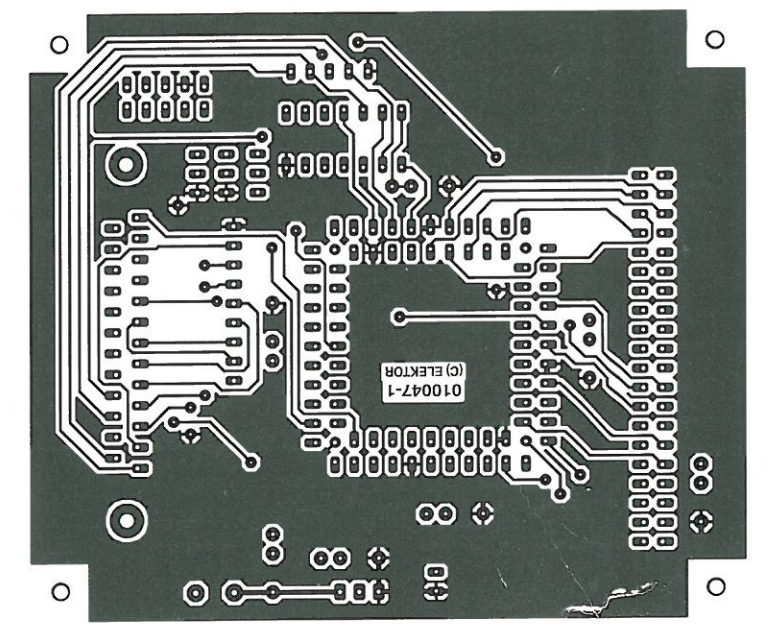
**HWR / I/O write** señal de escritura para las direcciones del puerto de I/O (E/S). En el flanco de bajada de estas señales, el contenido del bus de datos (HD0 a HD7 o HD0 a HD15) se transfiere hacia los registros del disco duro.

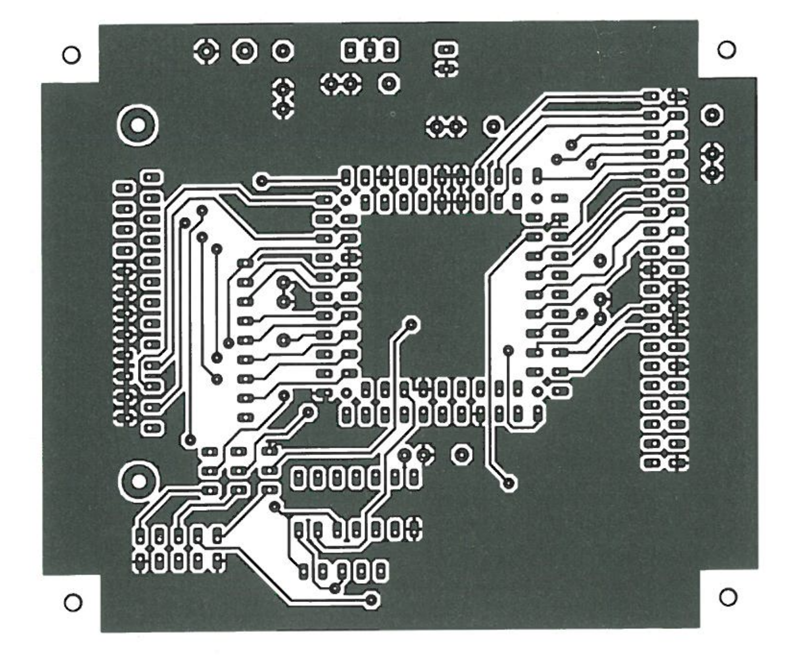
**HRESET / Reset** un nivel bajo en esta señal inicializará el disco duro.

El esquema eléctrico de la Figura 1 muestra el diagrama de tiempos para los ciclos de lec-

tura y escritura en el bus IDE. En el otro lado de la interfaz, tenemos el puerto paralelo de impresora que utiliza un conector Sub-D estándar de 25 terminales. El puerto de impresora tiene un bus de datos con ocho bits de ancho (desde el PC a la impresora) y cinco señales de estado de la impresora (desde la impresora hacia el PC). El dato se envía hacia el puerto LPT1 escribiendo en la dirección 378h (sólo escritura). Por su parte, el registro de control está (de nuevo sólo en escritura) en la dirección 37Ah mientras que el registro de estado está en la dirección 379h (sólo lectura). En la Tabla 2 se proporciona una descripción completa de las direcciones del puerto de impresora, tanto para LPT1 como para LPT2.





## LISTA DE MATERIALES

### Resistencias:

R1 = 1 KΩ, array SIL de 8 resistencias
R2 = 470 Ω, array SIL de 4 resistencias

### Condensadores:

C1,C3-C8 = 100nF
C2 = 100µF electrolítico de 16 V, radial

### Semiconductores:

IC1 = 7064LC84-15 (Altera), programada con código de pedido **010047-31**
IC2 = 74HCT14N
IC3 = 7805

### Varios:

JP1,JP2,JP3 = conector "pinheader" de tres terminales con puente de configuración
K1 = conector Sub-D de 25 terminales macho para montaje en placa de circuito impreso
K2 = conector "boxheader" de 10 terminales
K3 = conector "boxheader" de 40 terminales
2 terminales de soldadura
Cable de extensión para fuente de alimentación de PC (ver texto)
Cable de extensión de impresora (1:1)
Cable IDE
PCB placa de circuito impreso con código de pedido **010047-1**

Figura 5. Diagrama de pistas y distribución de componentes de la placa de circuito impreso de doble cara.

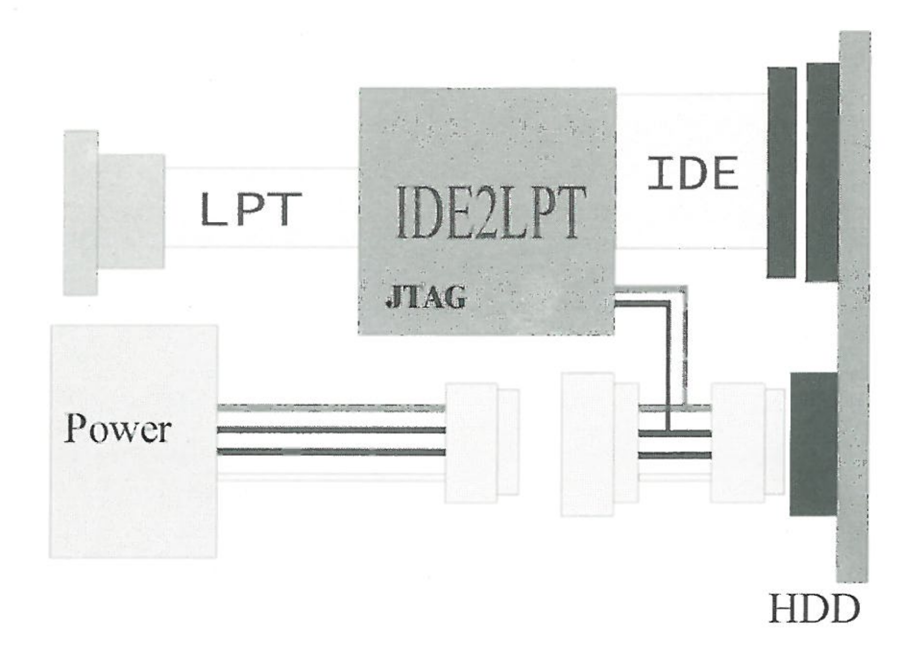



Figura 6. Cableado de la unidad junto con una fuente de alimentación de PC.

La velocidad de transferencia de datos del puerto de impresora es menor de 150 kBytes/s, pero tiene la ventaja de que todas las líneas de señales pueden controlarse por programa, lo que simplifica el diseño, tanto del programa de control como del circuito.

## El circuito adaptador

En el mundo de los discos duros la velocidad de transferencia de datos del puerto de impresora no es especialmente rápida. Sin embargo, hay que señalar que sí se puede hacer uso de todas las prestaciones y técnicas de mejora de velocidad que están disponibles en la interfaz IDE, tales como el Acceso Directo a la Memoria (DMA), una transferencia de datos de 16 bits, una situación de reposo y la posibilidad de diagnósticos. El interfaz adaptador de bus IDE a LPT es, prácticamente, un dispositivo amplificador y de almacenamiento entre los dos interfaces. En la Figura 2 se muestra el diagrama de bloques del circuito. El proceso de escritura en el disco duro se realiza en dos bloques, un bloque de ocho bits de ancho que se almacena en el registro de entrada (Reg In), y que se envía hacia el disco duro como palabras de 16 bits. En el sentido de lectura, el dato, con un ancho de 16 bits, se almacena en el registro de salida (Reg Out) y saca el valor leído en bloques de cuatro bits de ancho hacia el registro de estado del bus LPT, donde lo lee el ordenador. El control de la transferencia de datos se realiza a través de instrucciones enviadas hacia el registro de comandos (Reg Com), los cuales se listan en la Tabla 3. Se utilizará una lógica de control adicional para evitar situaciones prohibidas en el bus IDE (ver Tabla 4).

La mayoría de la circuitería del adaptador LPT/IDE ha sido implementada en un circuito integrado CPLD. Esto conlleva la definición del circuito completo, incluidos los registros y la electrónica lógica, con el Lenguaje Descripción Hardware, AHDL. A través del Servicio de Lectores de Elektor Electrónics se encuentra disponible un circuito integrado lógico preprogramado de la casa Altera, de la familia EPM 7064. Además, también está disponible el fichero fuente para el circuito integrado en la página web de Elektor Electrónics, que puede "bajarse" gratuitamente. En la Figura 3 se muestra el esquema eléctrico del circuito interno de este integrado, producido por este programa. La Figura 4 ilustra el esquema eléctrico de la placa adaptadora, donde IC1 es el circuito integrado CPLD. Además de este circuito integrado también está IC2, un buffer inversor 74HCT14, que dispone de unas entradas con histéresis que proporcionan una inmunidad al ruido mejorada en las entradas RHWR, RLWR, RCWR y HRESET. El circuito integrado IC3 proporciona una tensión de alimentación de + 5 V, necesaria para alimentar la tarjeta adaptadora, a partir de una fuente de alimentación externa de + 12 V (ver apartado "Conexión del conjunto"). El circuito integrado IC1 usa el conector K2 (JTAG) para permitir que dicho integrado pueda ser re-programado, así como realizar nuevas depuraciones. El programa "MAX2plus" para el circuito integrado está disponible en la página web de Altera (www.altera.com) para bajarlo gratuitamente, así como el programa para la programación del circuito integrado (llamado "BitBlaster"), aunque para conseguir este último hay que pagar unas pequeñas tasas. Si embargo, no es necesario comprar la aplicación BitBlaster o utilizar el conector K2, ya que el circuito integrado IC1 está disponible ya preprogramado a través del Servicio de Lectores de Elektor. Cualquier dispositivo CPLD de la familia 7000 de la casa Altera con el sufijo "S" (en nuestro caso es el componente EPM 7064S) puede usar el conector JTAG. Las tres señales de entrada para programar el circuito integrado IC1 (TCK, TMS y TD1), deben ser puenteadas a masa, a través de JP1, JP2 y JP3, para un funcionamiento normal del circuito, pero para realizar nuevas depuraciones y volver a programar el circuito integrado, los puentes deben

## Nota:

La interfaz descrita en este artículo se construyó sobre una placa prototipo (ver fotos) y fue verificada con un gran número de ordenadores y discos duros. Las diferentes pruebas se ejecutaron completamente sin problemas. Justo después de la fecha de cierre para publicación de este artículo, hemos recibido información del autor donde se indicaban posibles problemas con la temporización de algunos ordenadores más antiguos. De acuerdo con el autor, esto sólo sucedía en contadas ocasiones y en placas bases que contenían el microprocesador Pentium I, y dependiendo del conjunto de circuitos integrados montados para los puertos de E/S.

ser levantados y conectar estas señales a través del conector K2.

La placa del circuito impreso de doble cara que se muestra en la Figura 5 puede pedirse a Elektor Electonics. Si disponemos de los medios para fabricar nuestra propia placa de circuito impreso, podemos conseguir, de forma gratuita, los ficheros necesarios en la página web de Elektor Electonics.

## El programa

Al igual que sucede con cualquier puerto de ordenador, se necesita un programa controlador para pasar información, tanto de entrada como de salida, a través del adaptador LPT/IDE. Se dispone de dos versiones diferentes del programa controlador "LPT2IDE", una que corre bajo DOS y otra que se ejecuta bajo Windows. Así, en primer lugar, el programa "ide2lpt.exe" se ejecuta bajo MS-DOS a partir de la versión 3.3 de dicho sistema operativo. No existen demasiadas diferencias entre las rutinas del controlador IDE estándar, por lo que hemos adaptado un controlador IDE existente para leer y escribir a través del puerto de impresora. También se han incluido prestaciones adicionales de verificación y de búsqueda. Además, debemos de dar las gracias a Ewgenij Kuleschov por el código fuente del controlador. Para instalar el controlador se debe de añadir una línea en el fichero "config.sys":

```
Device = [senda]\ide2lpt.exe [opciones]
```

La instrucción más común será:

```
Device = c:\lpt2ide\i214.exe
```

Hubo verificaciones para diferentes configuraciones. Se pueden añadir distintas opciones al nombre del programa en la línea de comandos del DOS, como por ejemplo la opción /T (I2L4.EXE/T) para verificación.

Otras opciones disponibles son:

/H Ayuda.

/L:LPT Direcciones para el puerto LPT (378h para LPT1 y 278h para LPT2).

/G:SEC:HEAD defiende los parámetros del disco duro (utilizado para discos duros que trabajan en modo LBA). ":SEC" indica el número de sectores, mientras que ":HEAD" indica el número de cabezas. El formato se introduce en modo decimal.

/LLBA modo LBA.

/V solicita los parámetros del disco duro.

El controlador para MS-DOS utiliza particiones del tipo FAT 12, FAT 16, BIG y partición extendida, mientras que la partición FAT 32 viene soportada por el segundo controlador que se ejecuta bajo Windows 9x.

La instalación del controlador para Windows 95 se inicia asegurándonos primero de que disponemos de los controladores almacenados en el disco flexible. Una vez que hemos arrancado Windows, seleccionamos "Inicio" -> "Configuración" -> "Panel de control" -> "Añadir nuevo hardware". Seguidamente, en la aplicación de la instalación seleccionamos "Siguiente" y después "No", ante la pregunta: "¿Desea que Windows busque el nuevo hardware?". El paso siguiente será pulsar sobre "Siguiente" y buscar en la lista de controladores de "Tipos de hardware", picando sobre "Dispositivos SCSI". Pulsaremos de nuevo en "Siguiente" y seleccionaremos "Desde disco" y la opción "Mostrar". Navegaremos en el contenido del disquete hasta que encontremos el directorio donde están almacenados los controladores, seleccionando los ficheros ide2lpt.inf e ide2lpt.mpd. Finalmente, pulsaremos "Aceptar" y verificaremos que se inicia de forma automática el proceso de instalación. Antes de que podamos utilizar el dispositivo será necesario reiniciar nuestro ordenador.

Si alguna otra aplicación que se ejecuta bajo Windows accede al puerto de impresora mientras se están transfiriendo datos desde o hacia el disco duro, se pueden producir problemas. Por lo tanto, se recomienda que ningún otro programa cargado sobre nuestro ordenador pueda acceder al puerto de la impresora mientras se está ejecutando el programa IDE2LPT.

## Conexión del conjunto

La Figura 6 nos muestra todas las interconexiones entre las distintas partes que componen el adaptador completo. Se puede utilizar un cable de impresora estándar para conectar la unidad de disco duro portátil al puerto LPT de nuestro ordenador. Otra alternativa es utilizar un cable fabricado por nosotros mismos, usando cable plano con conectores IDC y Sub-D, respectivamente, en cada uno de sus extremos. Lo ideal es que la longitud del cable no supere los 30 cm. Si tenemos sospechas de que existen fuentes de radiofrecuencia (RFI) que nos puedan interferir y producir problemas en nuestra transferencia de datos, se pueden mejorar las ocho líneas de datos, DATA0 a DATA8, conectando condensadores de desacoplo (con valores comprendidos entre 12 y 22 pF), entre cada línea de datos y masa, en el conector adaptador K1.

Se recomienda obtener la alimentación para el disco duro y la placa adaptadora a través de una fuente de alimentación para ordenador independiente. Estas unidades proporcionan la tensión de alimentación necesaria con líneas de corriente independientes. No debe suponer ningún problema encontrar este tipo de fuentes de alimentación, sobre todo si estamos caníbalizando un viejo ordenador. Otra opción alternativa puede ser la de conseguir una fuente de alimentación nueva, cuyo precio no es excesivo. Las conexiones de alimentación entre el disco duro y la unidad de alimentación se pueden realizar con un cable de expansión de alimentación interno y estándar del ordenador, de donde sacaremos dos cables adicionales para proporcionar la tensión de + 12 y 0 V para la placa adaptadora. Estos dos cables deben ser soldados a los terminales etiquetados como "+" y "0", cerca del condensador C1, en la placa del circuito impreso.

## Disponer de gigabytes

Tanto si hemos decidido comprar un nuevo disco duro para este proyecto, como si reciclamos un viejo disco duro de un ordenador antiguo, el resultado es que este diseño tiene pocas exigencias para los ordenadores donde irá conectado (tanto si tienen como si no tienen unidades CD-ROM o CD-RW), permitiendo almacenar, realizar copias de seguridad o transferir ficheros de gran tamaño desde un ordenador a otro.

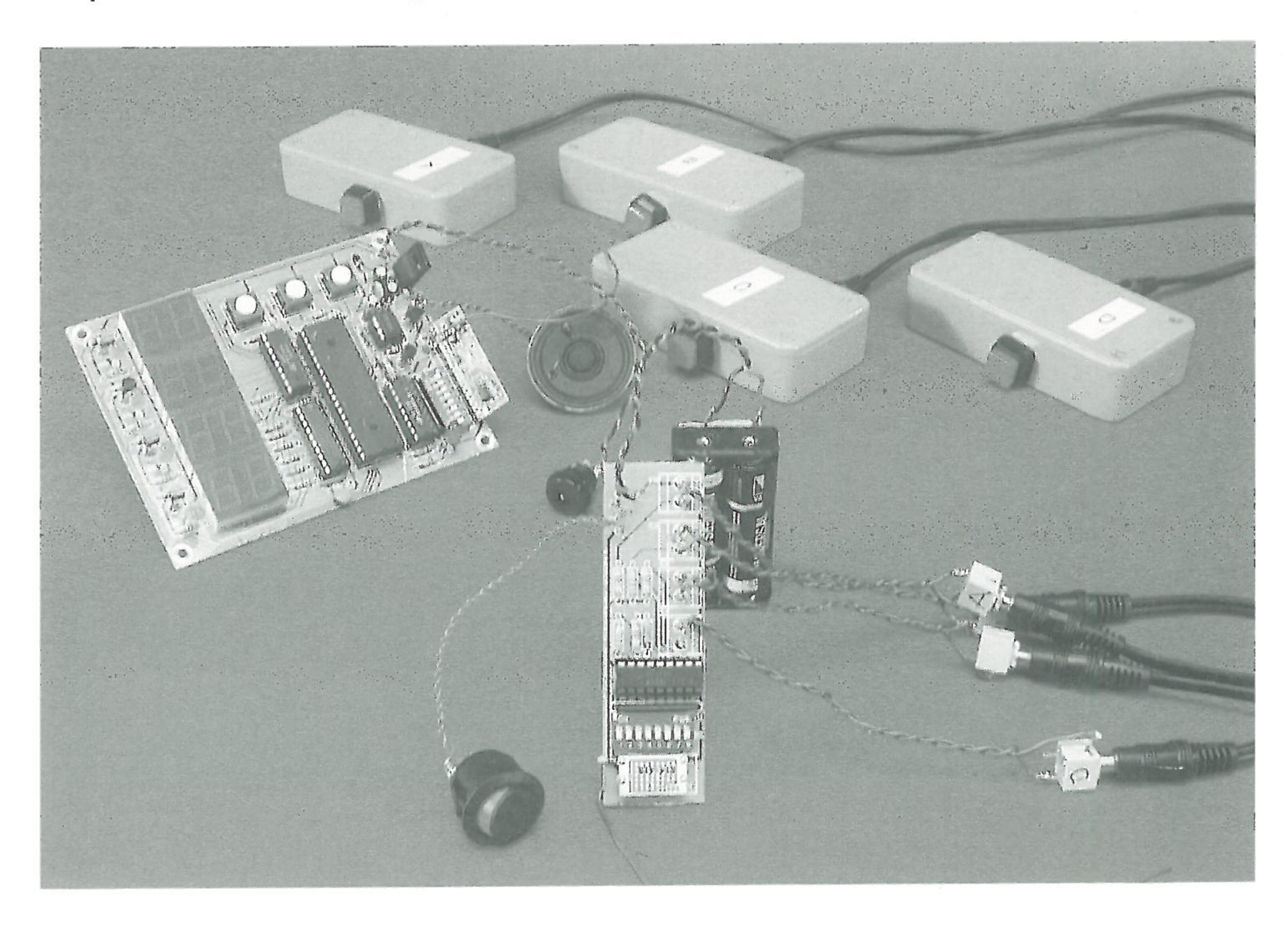
(010047-1)

# Interrogador maestro

## enlace vía radio!

Por Andy y Rose Morrell

**El interrogador maestro que presentamos en este montaje es una versión de cuatro jugadores con un jugador extra que controla la "caja" y que realiza las preguntas. No se necesita cableado entre el jugador y el dispositivo del interrogador maestro.**



La variedad de juegos basados en el conocimiento de la "Universidad Challenge", son verdaderamente populares y se han ramificado en varios juegos derivados que se pueden ver en televisión cada día. A pesar de que mucha gente sería perfectamente capaz de organizar un concurso de preguntas, convencer a los jugadores y al interrogador maestro para participar, generar una lista de cuestiones (en proporción al interés específico de los contendientes o al nivel de educación), concertar un punto de encuentro adecuado, construir una máquina para detectar la

primera tecla pulsada, mostrar el resultado del marcador, etc., a menudo supone una inmensa cantidades de problemas. El diseño que aquí proponemos debe ser fácil de montar para cualquier persona con una razonable destreza en el montaje de circuitos electrónicos.

Uno de los mayores problemas en el montaje de un "estudio" para preguntas, es el cableado. El proyecto interrogador maestro que se describe aquí nos ayuda a superar esta dificultad por medio de un enlace sin hilos entre el terminal de los contendientes y el terminal del maestro interrogador.

Pasemos a mostrar ahora cómo trabaja el sistema. Inmediatamente después de que el maestro interrogador realiza la pregunta, éste activa el botón de inicio del temporizador. El jugador que active su botón en primer lugar será bloqueado instantáneamente sobre el resto, de manera que si posteriormente a este hecho alguien más pulsa los botones, no tendrá ningún efecto. Para indicar qué jugador ha activado su pulsador en primer lugar, el indicador de diodos LEDs del correspondiente jugador parpadea y se escucha una sirena.

Si ninguno de los jugadores activa su pulsador, se alcanzará un tiempo máximo de fin de respuesta en, aproximadamente, 10 segundos. En esta situación, los cuatro indicadores LEDs comenzarán a parpadear simultáneamente hasta que el interrogador maestro presione el botón etiquetado como "Reset del temporizador".

Si se ha producido la respuesta correcta por parte de uno de los jugadores, todo lo que el interrogador maestro tiene que hàcer es pulsar el botón "correcto" y añadir 1 al indicador del jugador correspondiente o, si la respuesta ha sido errónea, dejar el resto de los marcadores inalterados. Seguidamente, se lanzará la siguiente pregunta.

El proyecto del interrogador maestro proporciona un fácil uso para los jugadores así como para el interrogador maestro, lo que permite que todo el mundo esté concentrado en las cuestiones, algo que es esencial para el éxito global del concurso si disponemos de la presencia de una audiencia. Los concursos nocturnos en un bar local, a menudo muy divertidos, quedan fuera de este tema.

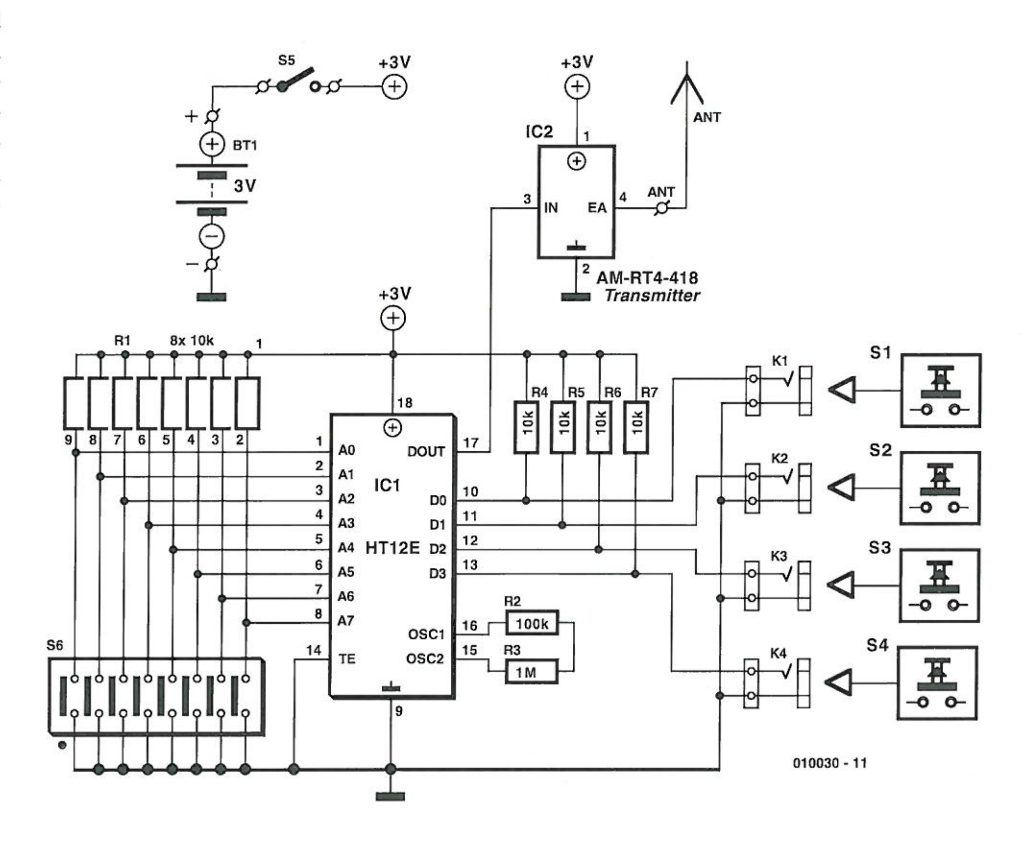



Figura 1. Esquema eléctrico del circuito del transmisor central del interrogador maestro, que tiene que ser montado en el interior, o muy próximo, de los terminales de los jugadores.

## Cómo trabaja

Al igual que sucede con las unidades emisoras y receptoras tratadas en otros artículos de este número, en este montaje existen otros dos componentes que siempre encontraremos en parejas. Así, los circuitos integrados HT12D y HT12E, de la casa Holtek que ya han sido utilizados en un proyecto anterior de Elektor, son el ejemplo de referencia. En la página web [1] del fabricante podemos encontrar las hojas de características de estos circuitos integrados. El esquema eléctrico del transmisor no es demasiado complejo, como se puede ver en la Figura 1.

El circuito integrado codificador HT12E, IC1, tiene sus cuatro entradas D0–D3 conectadas a los cuatro pulsadores en un pequeño alojamiento individual, a través de los conectores jack de 2,5 mm y de los terminales. El HT12E tiene su terminal de transmisión habilitada (TE) conexionado a masa, con lo que estará transmitiendo constantemente cuando se le conecte la alimentación (esto no afectará al receptor). El circuito completo de emisor y codificador funciona con dos pilas del modelo AAA o AA. La salida del circuito integrado codificador esta cableada hacia el transmisor de 418 MHz (o 433 MHz), que trabaja sin problemas con la misma tensión de batería de 3 V. Un trozo de hilo de 17 cm, utilizado como antena aérea, es suficiente para establecer las comunicaciones.

La comunicación (en un sentido) entre el HT12E y el HT12D está codificada. El mismo código de ocho bits (palabra) tiene que estar configurado en el bloque de conmutadores DIP (S5 y S6) para que el circuito receptor pueda reconocer al transmisor.

Toda la parte de inteligencia y control necesaria para gestionar la circuitería del interrogador maestro está agrupada en un único microcontrolador tipo PIC. Como podemos ver en el esquema eléctrico del circuito receptor de la Figura 2, no se necesita mucho más para que el integrado PIC sea capaz de manejar todas las funciones de entrada/salida, donde una "entrada" significa realmente la lectura de los datos provenientes del receptor de 433 MHz, además de la vigilancia de los botones de los terminales del interrogador maestro; mientras que la

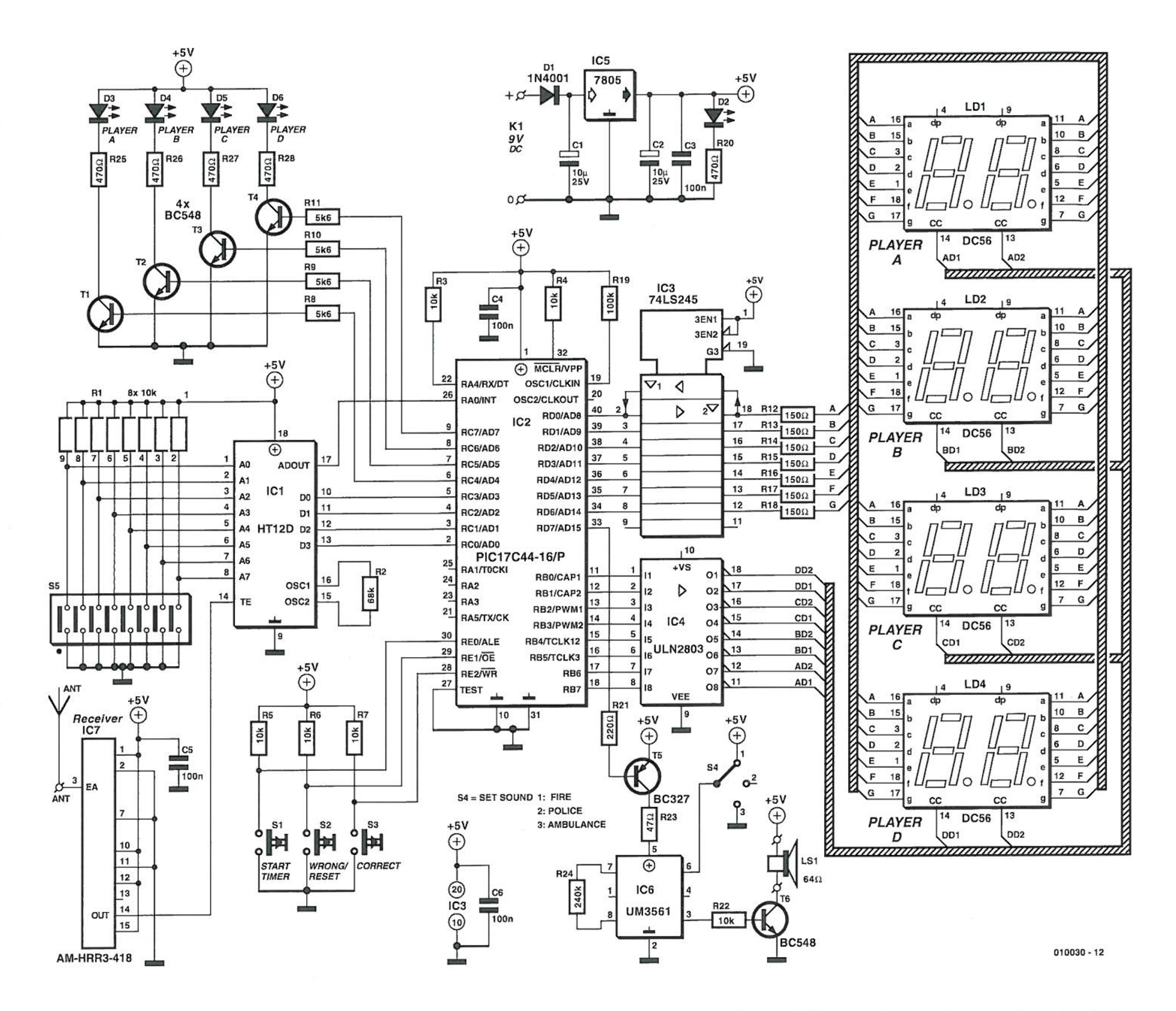



Figura 2. Esquema eléctrico del circuito del receptor central del interrogador maestro. Con un PIC en el sistema de control, la circuitería se reduce al mínimo.

salida se hace más pequeña para permitir el alojamiento de la sirena y los controladores de los indicadores LED y visualizadores. Una fuente de alimentación típica de 5 V completa el circuito.

Al presionar el botón "START" se ejecuta la rutina que permite al PIC reconocer si el circuito integrado receptor HT12D, recibe una señal proveniente de los jugadores A-D. El circuito integrado PIC actualiza el contador del programa creando una interrupción en cada milisegundo. Si no se recibe ningún dato desde el transmisor central (por ejemplo, cuando ninguno de los jugadores conoce la respuesta a la pregunta), el contador generará un periodo de temporización que dura unos 10 segundos. En ese momento el integrado PIC hará parpadear los cuatro indicadores LEDs denominados "Jugadores A–D" (D3–D6), hasta que el interrogador maestro presione el botón de WRONG/RESET (es decir, RESET o ERROR).

Cuando se produce una respuesta correcta dentro del espacio de tiempo disponible de aproximadamente 10 segundos, el interrogador maestro usará el botón "CORRET" (CORRECTO). El circuito integrado que activa la sirena, IC6, está equipado con sonido. En este caso, la línea del terminal del puerto del PIC, RD/, está seleccionada a nivel bajo, de manera que, a través del transistor PNP BC 327 (T5) se activa el circuito integrado de la sirena, un UM 3561. El transistor BC 548, situado a la salida del circuito integrado UM 3561, permite controlar un pequeño altavoz de 64Ω, de manera que cuando suene la sirena se oirá perfectamente. En el mercado existen tres tipos de sirenas de sonido muy conocidas, que pueden seleccionarse a través del conmutador S4.

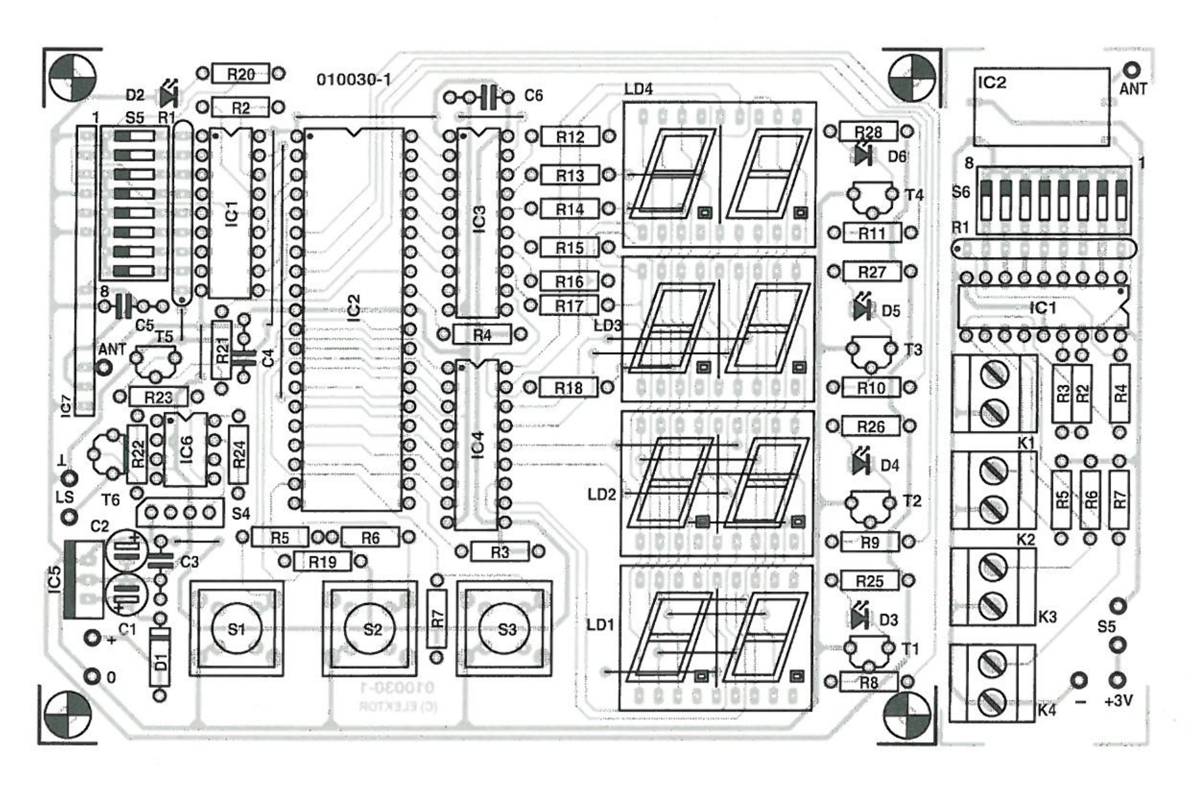
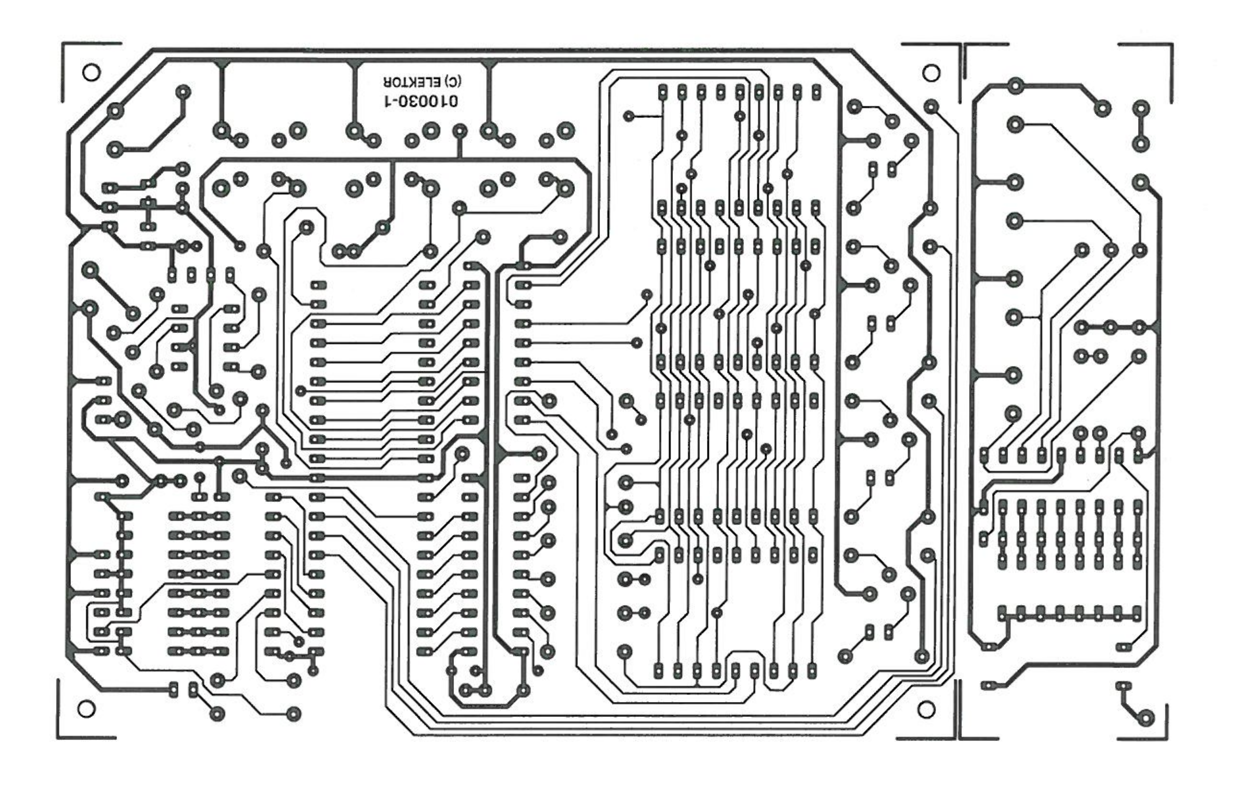



Figura 3. Distribución de pistas de cobre y plano de montaje de componentes de la placa combinada emisora y receptora (separadas por su línea de corte antes del montaje de los componentes). Esta placa está disponible ya montada y verificada.



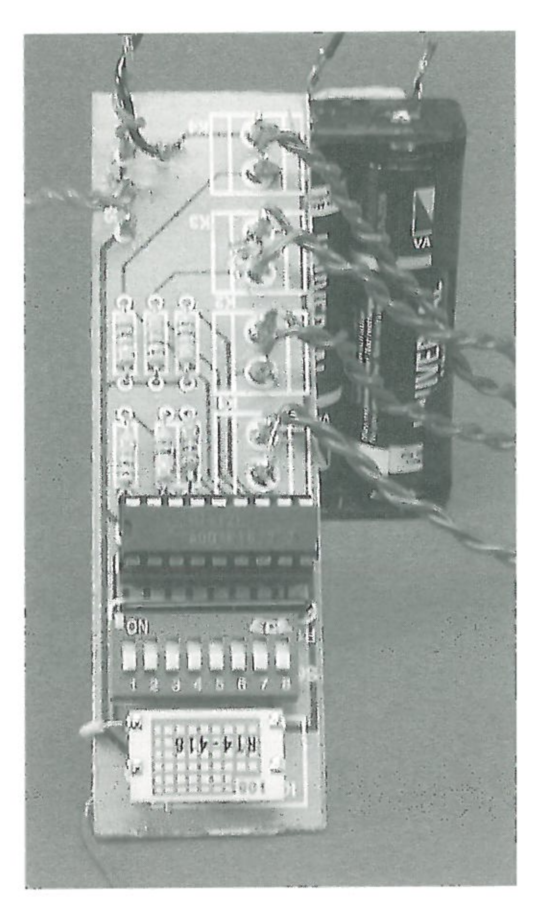


Figura 4. Placa completa del transmisor.

## LISTA DE MATERIALES
### Transmisor

**Resistencias:**
R1 = 10 KΩ, array SIL de 8 resistencias
R2 = 100kΩ
R4..R7 = 10kΩ
R3 = 1MΩ

**Semiconductores:**
IC1 = HT12E (Farnell # 562415)
IC2 = AM-RT4-433 (RF Solutions)*

**Varios:**
K1-K4= Conector jack de 2,5 mm, mono, para montaje en placa de circuito impreso
S1-S4 = Pulsadores (externos) de un solo contacto
S5 = Conmutador de encendido/apagado
S6 = Conmutador DIP de 8 líneas
BT1 = Batería de 3 V con soporte (o dos baterías de tipo AA/AAA).

### Receptor

**Resistencias:**
R1 = 10 KΩ, array SIL de 8 resistencias
R2 = 68kΩ
R3-R7,R22 = 10kΩ
R8-R11= 5kΩ6
R21 = 220Ω
R12-R18 = 150Ω
R19 = 100kΩ
R20 = 470Ω
R23 = 47Ω
R24 = 240kΩ
R25-R28=470Ω

**Condensadores:**
C1,C2 = 10 μF, electrolítico de 63 V, radial
C3-C6 = 100nF

**Semiconductores:**
D1 = 1N4001
D2-D6 = diodo LED verde de 5 mm de diámetro
IC1 = HT12D (Farnell # 563250)
IC2 = PIC 17C44-16/P programado, con código de pedido **010030-41**
IC3 = 74LS245
IC4 = ULN2803AP (Farnell # 3187032)
IC5 = 7805
IC6 = UM3561 (UMC) (Rapid Electronics # 82-0704, tel. 01206 751166)
IC7 = AM-HRR3-433 (RF Solutions) *
LD1-LD4= visualizador LED de 2 dígitos y 7 segmentos, con cátodo común (por ejemplo, Kingbright DC56-11EWA)
T1-T4,T6 = BC548
T5=BC327

**Varios:**
K1 = Conector de adaptador de red para montaje en placa de circuito impreso
S1,S2,S3= Pulsadores de un solo contacto, tipo CTL3 (Multimec) o DR-6
S4 = Conmutador de un circuito tres posiciones
S5 = Conmutador DIP de 8 líneas
LS1 = Altavoz de 64 Ω
PCB Placa de Circuito Impreso con código de pedido **010030-1**. (Placa combinada para emisor y receptor)

* 433 Módulos receptor y emisor disponible como par de elementos en la casa Maplin, con código de pedido VY48C



Figura 5. Placa completa del receptor. ▽

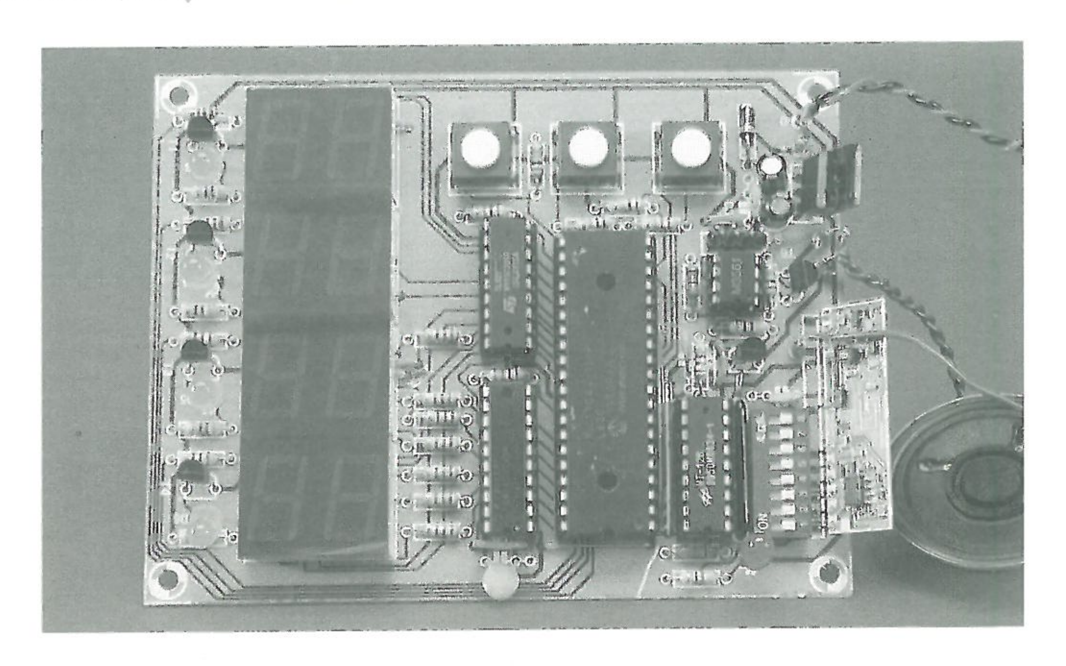
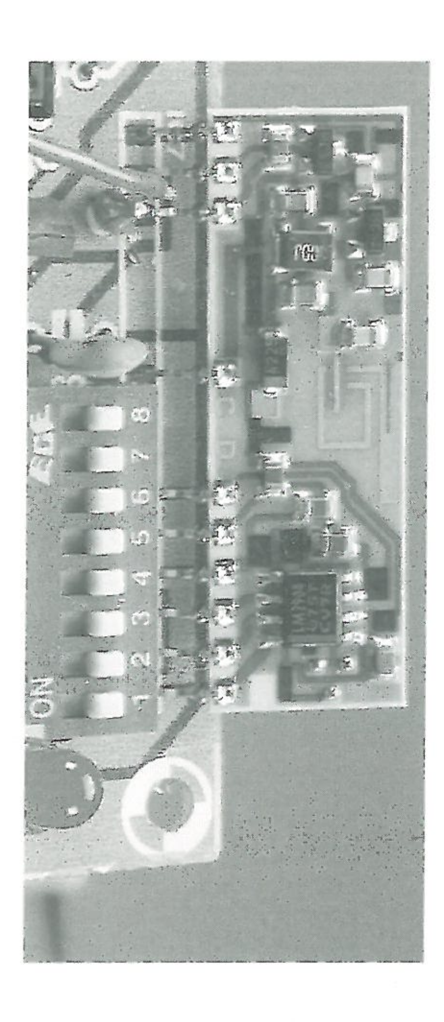


Figura 6. Detalle del módulo receptor de radio RF Solution AM-HRR3-433, montado sobre la placa principal.



Como ya hemos mencionado anteriormente, la comunicación entre los distintos dispositivos de los jugadores y el dispositivo de control del interrogador maestro, se realiza sin hilos. En este montaje se utiliza un par de emisores/receptores de bajo coste y exentos de licencia, de la casa RF Solutions. Las unidades que el autor utilizó originalmente funcionaban en la banda de 418 MHz. Un dispositivo exento de licencia significa que el transmisor puede ser utilizado por cualquier persona libremente, con tal de que su circuitería no sea modificada o conectada a una antena direccional. También existen las versiones del transmisor y del receptor de 433 MHz, ya que en el Reino Unido la frecuencia de 433,92 MHz también se utiliza para comunicaciones libres de licencia de corto alcance. El uso de la banda de 433,92 MHz se ha establecido por toda Europa, por lo que se espera que el modelo de 418 MHz se quede, de forma gradual, desfasado en el Reino Unido. En Internet [2] podemos encontrar las hojas de características de los módulos AM-RT4-418/433 y AM-HRR3-418/433.

Volviendo a la unidad principal instalada en el terminal del interrogador maestro, el circuito integrado PIC 17C44 controla las unidades de visualización en modo multiplexado, a través de dos circuitos integrados controladores, un ULN 2803 para las líneas de cátodo común (CC) y un 74LS245 para las líneas de los dígitos (ánodo). El montaje dispone de cuatro visualizadores de dos dígitos cada uno y de siete segmentos por cada dígito. Cuatro transistores BC 548 (T1-T4) controlan los indicadores LEDs de 5 mm de cada jugador.

Para finalizar la descripción del funcionamiento del circuito, la fuente de alimentación está basada, como suele ser tradicional, en el regulador de tensión de tres terminales 7805, que disminuye la tensión continua de entrada no regulada (max. de 9 V) presente en K1, a los 5 V requeridos por la circuitería del receptor. El diodo LED D2 actúa como el indicador de encendido y apagado. Pensando en la seguridad eléctrica, recomendamos el uso de un adaptador de tensión de red con un rango de corriente de salida de unos 300 mA o superior.

## El programa de control

El fichero con el código fuente para el programa que se ejecuta dentro del circuito integrado PIC es demasiado largo para ser listado en este artículo, pero se puede conseguir solicitando el disquete que lo contiene, con el código de pedido 010030-11 (con gastos para el solicitante), o bajarlo gratuitamente de la página web de Elektor Electronics. Si dispone del correspondiente programa ensamblador del PIC, puede utilizar el fichero de código ensamblado para generar su propio código objeto y grabar su propio PIC 17C44 para este proyecto. Las líneas de comentarios nos pueden ayudar fácilmente a entender el funcionamiento del programa y representan un valor educativo importante, incluso si no tiene pensado construir el interrogador maestro. También está disponible el circuito integrado PIC ya programado con el código de pedido **010030-41**.

## Montaje

El transmisor y la unidad principal se montan sobre las placas de circuito impreso cuyo diagrama de pistas y de implantación de componentes se muestra en la Figura 3. Debemos señalar que las dos placas deben separarse antes de montarlas sobre su caja. Las fotografías de las Figuras 4, 5 y 6 nos dan una buena idea de cómo se deben montar los componentes sobre estas placas.

La placa del receptor contiene la mayoría de los componentes, incluyendo los visualizadores, y debe montarse en una posición en la que las lecturas presentadas (puntuación conseguida e indicación de los diodos LEDs de los jugadores) se puedan leer claramente por cada uno de los jugadores. Esto puede exigir montar los botones de control del interrogador maestro fuera de la placa, en una lámina de panel independiente.

Recomendamos utilizar zócalos para todos los circuitos integrados de las placas. Sobre todo para el circuito integrado PIC ya programado que conlleva un valor considerable.

Las antenas del transmisor y del receptor son, en ambos casos, unos trozos de hilo rígido de unos 17 cm que, para una óptima radiación (en el transmisor) y recepción (en el receptor), no deben incluirse en una caja de metal.

El regulador de tensión IC3 no puede funcionar a temperaturas excesivas, por lo que, como medida de seguridad, podemos montar sobre él un pequeño radiador.

El altavoz de 64 Ω puede instalarse en una caja externa o en un espacio específico en la caja central del receptor.

Podemos utilizar cualquier código de acceso que deseemos para codificar el transmisor, con la condición de que seleccionamos el mismo código en el receptor y de que dicho código sea exclusivo del resto de los transmisores en un área de unos 300 m.

(010030-1)

### Direcciones Web

[1] Hojas de características de los circuitos HT12E/HT12D: www.holtek.com.tw
[2] Hojas de características de los módulos AM-HRR3-418/433 y AM-TR-418/433: www.rfsolutions.co.uk

# ISAC (4)

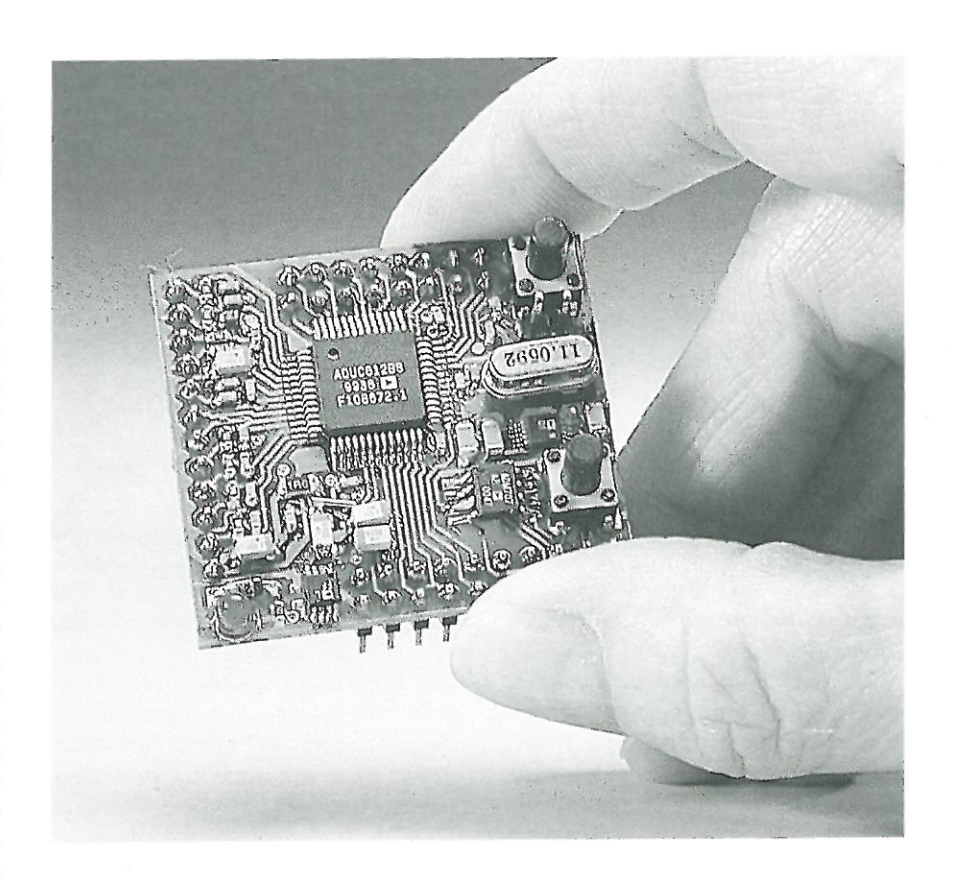
## parte 4: aplicaciones básicas

Por Prof. B. vom Berg, P. Groppe y M. Müller-Aulmann

**En esta cuarta entrega de la serie, volveremos a echar una ojeada sobre nuestras primeras aplicaciones del cubo ISAC y de la sencilla placa base de interconexión. Los programas ejemplo completos desarrollados para estas aplicaciones, además de una información extra, están disponibles para poderlos "bajar" de la página web de Elektor Electronics.**

Antes de presentar la aplicación ejemplo individual, debemos remarcar algunos detalles que nos ayudarán a usar correctamente los programas ejemplo.

- Las aplicaciones y los programas de demostración están escritos en código C51, con lo que (con una excepción) pueden emplearse con la versión limitada de µVision2 (tamaño de código máximo de 2 Kbytes y aritmética sin coma flotante). Para programas de aplicación de mayor tamaño debemos de proveernos de la versión completa del programa µVision2 o de otra herramienta para el 8051.
- Para poder trabajar con algo "visible" todos los programas dirigen la información de usuario y las distintas salidas generadas, hacia el monitor del ordenador. Esto significa que necesitamos ejecutar un programa terminal en nuestro PC. En el proyecto hemos supuesto que estemos utilizando la aplicación HyperTerminal de Windows, ya que este programa viene incluido con el propio sistema operativo. Sin embargo, podemos utilizar cualquier otro programa terminal: los parámetros necesarios para el establecimiento de la comunicación son: 9.600 baudios, 8 bits de datos, sin paridad y 1 bit de parada.
- Antes de ejecutar un programa en el cubo ISAC debemos ejecutar nuestro programa terminal seleccionado y pulsar el botón reset en el propio cubo.




- **Muy importante:** si hemos estado utilizando la aplicación HyperTerminal y queremos volcar un nuevo programa en el cubo ISAC utilizando dicha aplicación, lo primero que debemos hacer es eliminar la conexión existente con la aplicación HyperTerminal (utilizando la opción "Desconectar"). Si no lo hiciésemos de esta manera, el HyperTerminal mantendría el control del puerto COM correspon-

diente y el programa "cargador" serie no sería capaz de acceder al mismo. Además, aparecerá un mensaje de error, sin mucho sentido aparente, si el programa "cargador" serie es incapaz de encontrar el ADuC812. En este caso, tendremos que cancelar el mensaje de error, liberar la conexión en el HyperTerminal y volver a lanzar el programa "cargador".

- La comunicación entre el núcleo del microcontrolador y los otros circuitos integrados periféricos, utiliza el Registro de Función Especial (SFR) del 8051 (ver Figura 1). En las hojas de características del ADuC812 podemos encontrar información adicional del significado de los distintos bits en el interior del SFR. Otro lugar donde podemos encontrar esta información es en libros especializados sobre el 8051.

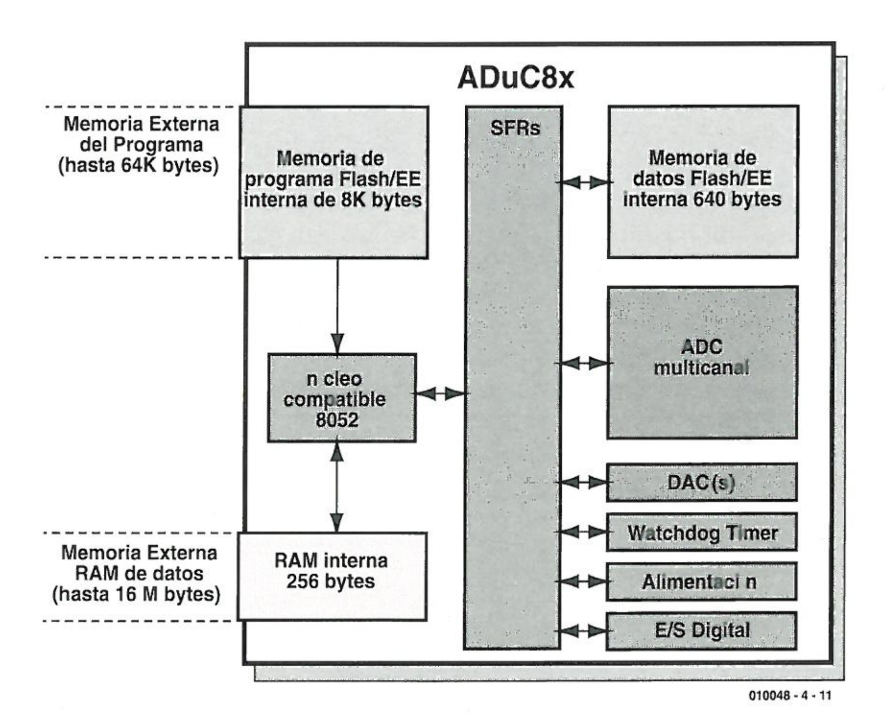



Figura 1. El registro SFR.

## Aplicación para el conversor D/A

**Obtención de diferentes señales de salida:**
Con la ayuda del programa **dac_1.c,** podemos colocar en las salidas dos señales diferentes (una onda cuadrada y un diente de sierra), utilizando el conversor D/A con dos resoluciones diferentes (de 8 y de 12 bits). Para verificar los resultados podemos conectar un osciloscopio a las correspondientes salidas.

La tensión de referencia utilizada en este caso es la tensión de la fuente de alimentación digital VDD (= + 5 V), por lo que las tensiones de salida producidas están comprendidas en el rango de 0 a + 5 V.

## Aplicación para el conversor A/D

Lectura de las tensiones en los canales de entrada:

Al ejecutar el programa adc_1.c se nos preguntará el número de canales que van a ser leídos, con lo que tendremos las siguientes posibilidades:

–Números de 0 a 7: canales de medida externos ADC 0 a ADC 7.

–Número 8: canal conectado al sensor de temperatura interno (ver ejemplo siguiente).

Cuando hemos introducido un número, la tensión presente en el canal de entrada seleccionado se convierte en una señal digital una vez por segundo y el resultado se mostrara en la pantalla de nuestro PC, como un valor decimal y como un valor hexadecimal. Las diferentes lecturas se detienen al pulsar el botón reset, con lo que, en ese momento, tenemos la oportunidad de elegir un nuevo canal de entrada.

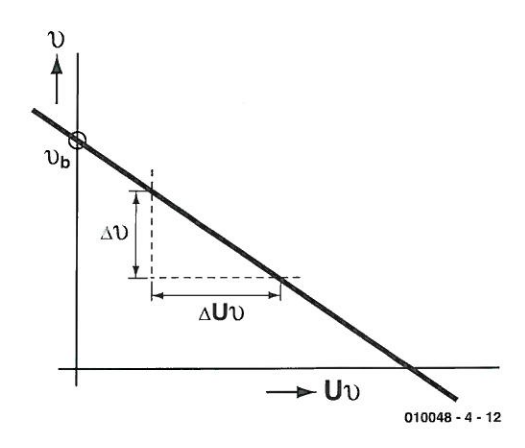



Figura 2. Relación lineal entre la tensión $U_{\vartheta}$ y la temperatura $\vartheta$.

**NOTA:** El conversor A/D tiene una resolución de 12 bits y puede funcionar con dos tensiones de referencia diferentes:

1 Utilizando la tensión de referencia interna de 2,5 V. En este caso el bit de menor peso LSB 1 se corresponde con un paso de tensión de 610 µV.

2Usando una tensión de referencia externa en el rango comprendido entre + 2,3 y + 5,0 V conectada al terminal VRef. Teniendo en cuenta que la resolución es de 12 bits, esta tensión debe ser filtrada y regulada a conciencia. Si, por ejemplo, la tensión VRef

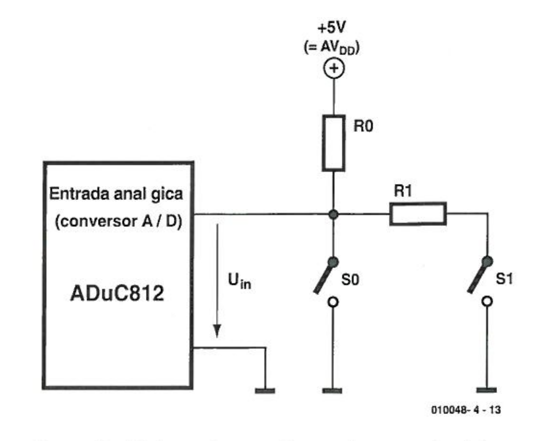



Figura 3. Divisor de tensión en la entrada del conversor A/D, conmutado por pulsadores.

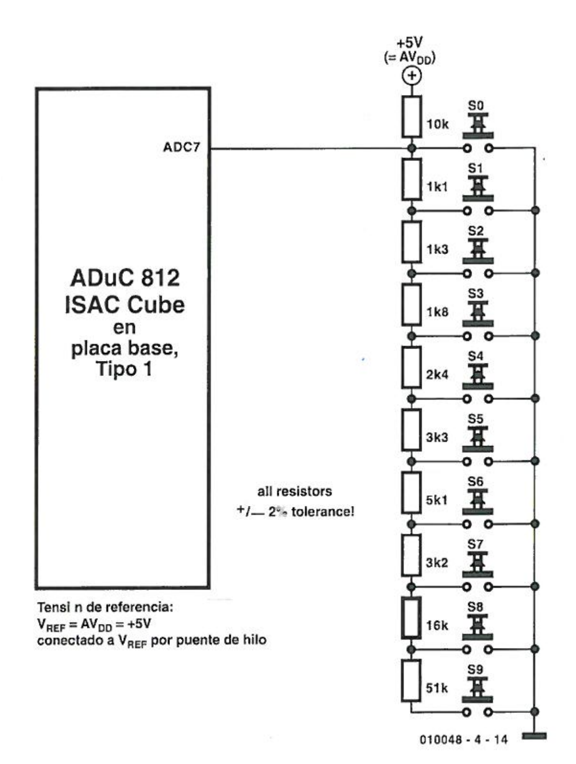



Figura 4. Serie de pulsadores conectados a un terminal de entrada analógico.

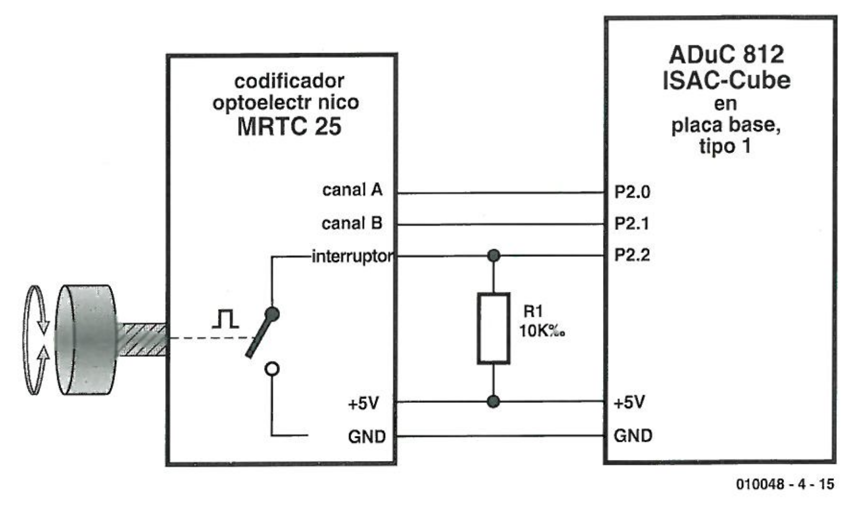



Figura 5. Conexión de un codificador flecha con electrónica óptica.

se conectara al terminal de alimentación AVDD (+ 5 V), el bit de menor peso LSB 1 se correspondería a un paso de tensión de 1,2 mV.

La tensión de entrada que va a ser medida siempre debe estar conectada entre 0 V y la tensión de referencia utilizada (tanto si es interna como externa) como por ejemplo, en el rango de 0 a 2,5 V, cuando utilizamos la tensión de referencia interna, o en el rango de 0 a + 5 V si estamos empleando una tensión de referencia externa de 5 V.

**Medida de la caída de tensión interna y la temperatura ambiente:**

El ADuC812 incluye un sensor de temperatura interno que produce una temperatura que depende de la tensión a la que está conectado. Esta tensión se dirige hacia el canal noveno (por ejemplo, canal número 8) del conversor A/D. Las hojas de características proporcionan la siguiente información sobre este sensor: la tensión de salida es de 600 mV a la temperatura de $\vartheta$ = 25° C, mientras que la dependencia de la temperatura es de - 3 mV/° C. Así, si la temperatura aumenta en 1° C, la tensión dependiente de la temperatura caerá en 3 mV y viceversa. Pero, llegados a este punto, hay un **detalle muy importante** a señalar: estos datos son solamente **valores típicos** y varían de un componente a otro. Si comparamos dos ADuC812 de diferentes líneas de producción, encontraremos unas diferencias de valores bastante importantes. Sin embargo, la casa Analog Devices garantiza que la característica de tensión dependiente de la temperatura es muy lineal con la variación de la temperatura.

Esta relación entre la tensión y la temperatura viene caracterizada por una ecuación lineal (ver **Figura 2**):

$$U_\vartheta = f1(\vartheta) \text{ (o } \vartheta = f2(U_\vartheta)).$$

Para el usuario, esto significa que se deben de tomar dos lecturas para realizar la calibración y poder establecer los parámetros de la

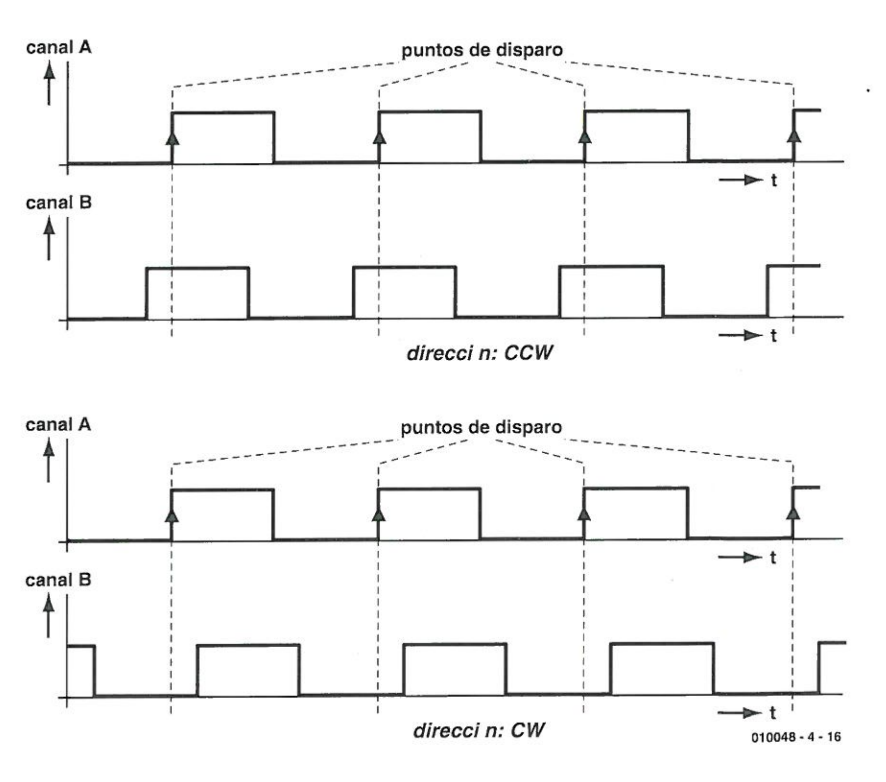



Figura 6. Diagrama de tiempos para las dos direcciones de rotación.

ecuación lineal, antes de que se comiencen a obtener los valores de temperatura medidos. Por lo tanto, si disponemos de dos ADuC812 de la misma línea de producción (lo que suele suceder cuando nos suministran el circuito integrado del mismo tubo de transporte), no necesitamos cambiar nada. Sin embargo, si comparamos diferentes ADuC812 de diferentes líneas de producción, el proceso de calibración tendrá que repetirse varias veces. En el peor caso, tendremos que obtener un conjunto de parámetros diferentes para conseguir obtener la relación lineal de cada ADuC812. El programa **temp812.c** ha sido desarrollado para ayudarnos en el proceso de calibración. Cuando cargamos el programa encontramos que disponemos de unas instrucciones que nos ayudarán a determinar los parámetros de la relación lineal, **midiendo la temperatura ambiente**, y devolviendo los parámetros programados hacia el programa temp812.c. Debemos seguir las instrucciones que se nos dan de cómo utilizar dicho programa.

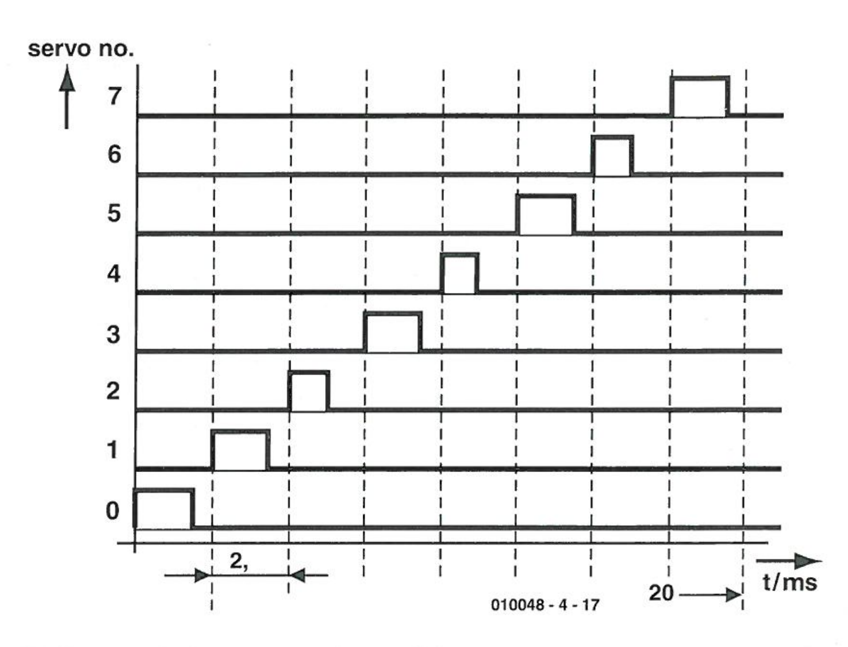



Figure 7. Diagrama de tiempos para el control de un servo.

**Pulsadores conectados a una entrada analógica:**

Las bases para este método de conexión de pulsadores, que nos permite conectar un teclado completo a un terminal de una entrada analógica del conversor A/D, es la de conseguir un "divisor de tensión con múltiples conmutaciones". Para comenzar, iniciaremos el ejemplo con un único divisor de tensión (ver **Figura 3**).

Con este circuito tendremos:

Caso de no pulsar ningún botón:

$U_{in} = 5$ V.

Caso del botón 0 pulsado:

$U_{in} = 0$ V.

Caso del botón 1 pulsado:

$U_{in} = 5\,[R1 / (R0 + R1)]$ V.

Si las dos resistencias, R0 y R1, tienen el mismo valor, por ejemplo, 1 kΩ, tendremos una tensión $U_{in}$ = 2,5 V.

Cada vez que pulsamos un nuevo botón (o tecla), nos aparece una tensión diferente y distinguible en la entrada analógica, que es convertida por el conversor A/D. De este modo, para cada botón que se pulse podemos obtener un valor digital diferente, por lo que podemos identificar claramente el botón pulsado. Hay otra ventaja adicional en este circuito: si se pulsa más de un botón de modo simultáneo, sólo el botón más próximo a la entrada analógica será reconocido. En otras palabras, podemos construir un teclado en el que las distintas teclas tengan una cierta prioridad. Si, por ejemplo, el botón 1 ha sido pulsado y, a continuación, se pulsa el botón 0, la tensión Uin cambiará inmediatamente a la tensión correspondiente al botón 0 (0 V). Lo que significa que la información del botón 1 se sobrescribe y ya no será tenida en cuenta, mientras que la tecla 0 tiene la prioridad más alta y siempre será reconocida. Cuando se suelta la tecla 0, pueden leerse el resto de las teclas.

Para utilizar un gran número de teclas se requiere que las resistencias que se usen tengan una tolerancia muy baja, al igual que la tensión de alimentación regulada del divisor de tensión. La **Figura 4** nos muestra un circuito práctico para un teclado con diez pulsadores.

Una pequeña desventaja de esta idea, y

## Tabla I

**Características del codificador en flecha optoelectrónico, MRTC25.**

**Características eléctricas:**

| | |
|---|---|
| Resolución (pulsos/360°) | 25 |
| Canales | 2 (A y B; desplazamiento de la fase 90°) |
| Tensión de alimentación | 5 V DC ±0.5 V |
| Consumo de corriente | 20 mA |
| Señales de salida | Señales cuadradas con niveles TTL, con resistencia interna de "pull-up" de 10 kΩ |
| Pulsadores | 5 V, 10 mA (con carga no inductiva) |
| Tiempo de vida de funcionamiento | > 1,000,000 ciclos |

**Fuente para los componentes:**
MEGATRON Bauelemente,
Hermann-Oberth-Strasse 7,
D-85640 Putzbrunn, Munich, Alemania.
Tel.: +49 89 46094 146

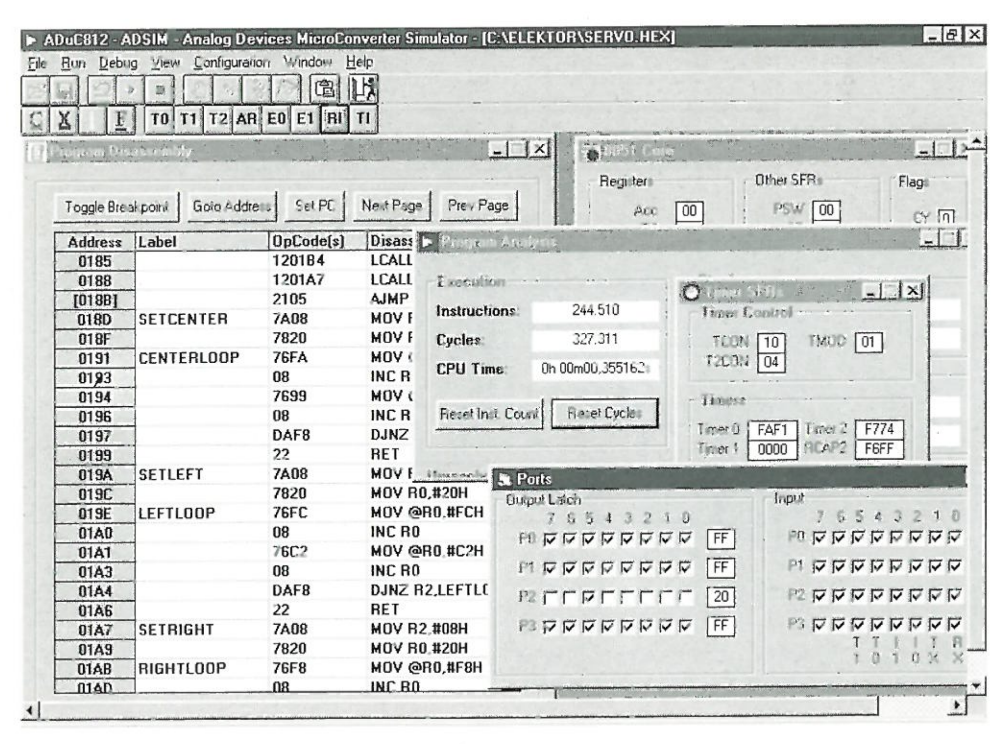



Figure 8. Fichero "servo.hex" en el simulador ADuC812 (la salida de los pulsos para el servo 5 está en el terminal P2.5).

que no debe ser ignorada, es que los botones tan sólo pueden leerse cuando están siendo pulsados: no existe un circuito o "buffer" de almacenamiento de teclado. Sin embargo, esta omisión puede rectificarse a través de un módulo de programa dedicado. En el fichero **ana_tast.c** podemos encontrar el programa que nos permite realizar la lectura de un teclado con diez teclas conectado al terminal ADC7 de la entrada analógica del ADuC812. Cuando cargamos este programa podemos ver que también disponemos de una tabla que nos muestra la relación entre la tecla pulsada, la tensión de entrada $U_{in}$ y el valor digital convertido, además de los intervalos de seguridad de lectura del pulsador y el nivel de prioridad de dicho pulsador (0 = prioridad más alta).

## Aplicación para el puerto de E/S digital

Uso de un codificador de flecha con circuitería óptica:

Con un codificador de flecha con circuitería electrónica conseguimos obtener un sencillo y elegante dispositivo de entrada. En nuestro proyecto utilizamos el modelo MRTC 25 de la casa Megatron, que ofrece dos canales de entrada (A y B) para conteo e indicador de dirección (en sentido horario y en sentido antihorario). También proporciona un contacto que se cierra cuando la flecha se pulsa hacia abajo.

La conexión con el cubo ISAC o a la sencilla placa base (tipo 1) se realiza como se muestra en la **Figura 5**. El diagrama de tiempos para la rotación en sentido horario y antihorario se muestra en la **Figura 6**. Las características eléctricas del codificador de flecha se proporcionan en la **Tabla 1**. El programa **drehge_1.c** nos muestra cómo realizar la interfaz con este interesante dispositivo de entrada.

**Control de ocho servos:**

Utilizando las ocho salidas digitales del puerto 2 y un pequeño programa, en este caso descrito en el ensamblador, es fácil llegar a controlar ocho servos. Para cada servo se debe generar un pulso de control de una duración de entre 0,9 y 2,1 ms, cada 20 ms (ver **Figura 7**). El método de aproximación más sencillo es dividir el período de 20 milisegundos en ocho intervalos más pequeños y, a continuación, controlar cada servo de manera sucesiva, desde la correspondiente salida del puerto 2 hacia la posición deseada. El programa **servo.asm** genera los intervalos de 2,5 ms, utilizando el Temporizador 2 en modo "carga automática", y generando las temporizaciones necesarias para los pulsos de control de los servos, mediante el temporizador 0 como un temporizador de 16 bits.

Intervalo: 20 ms / 8 = 2,5 ms (temporizador 2)
Ancho del pulso para el servo: de 0,9 ms a 2,1 ms (temporizador 0)

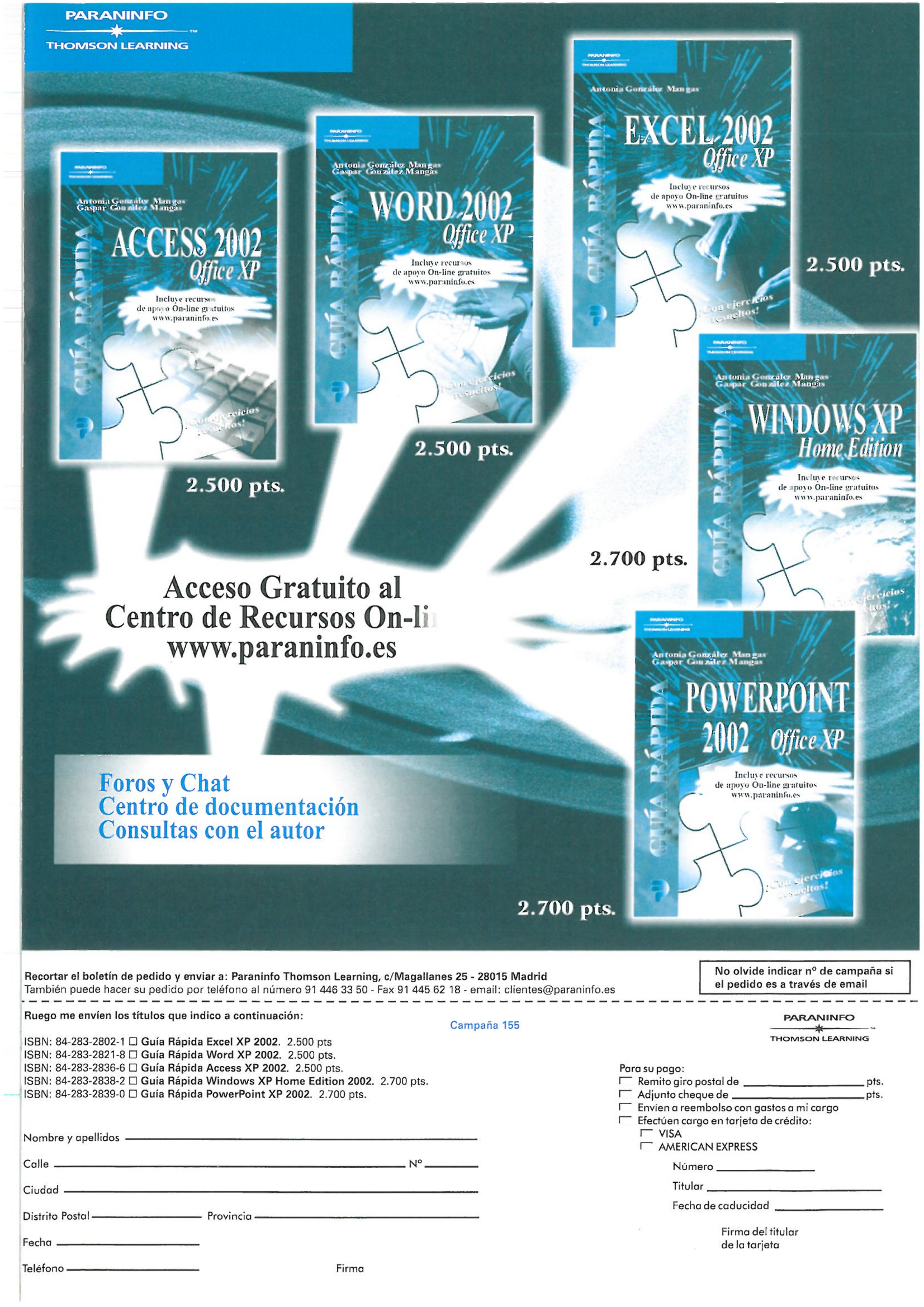
En el temporizador 2, la rutina de servicios de interrupción del temporizador 0 se inicia cuando se obtiene el ancho del pulso necesario para el servo y cuando se ha seleccionado el terminal correspondiente del puerto 2. En la rutina de servicios de interrupción del temporizador 0 todos los terminales del puerto 2 han sido borrados y el temporizador se ha detenido.

Para alcanzar la temporización exacta en el programa principal, se ha utilizado un pequeño programa de retardo adicional en la rutina de servicios de interrupción del temporizador 2. Este pequeño programa se puede probar rápidamente utilizando el simulador ADuC812 (ver **Figura 8**).

Está claro que las únicas limitaciones para las distintas aplicaciones que deseemos crear para el ADuC812, son las que ponga nuestra imaginación. Con un pequeño esfuerzo adicional en la parte de programación, podemos llegar a controlar hasta 24 servos utilizando las 24 salidas digitales. Cuando se usa en combinación con el interfaz I$^2$C, podemos llegar a construir incluso un sustituto para el "Interfaz de servos I$^2$C", con la ventaja de que nuestro sustituto proporciona una resolución de más de 1.000 pasos.

Los servos también podrían ser controlados por sensores a través de las ocho entradas analógicas. Es precisamente esta opción la que trata esta serie de artículos (recordemos que ISAC viene de Intelligent Sensor/Actuator Controller, es decir, Controlador Inteligente de Sensores/Activadores).

(010044-4)




**Recortar el boletín de pedido y enviar a: Paraninfo Thomson Learning, c/Magallanes 25 - 28015 Madrid**
También puede hacer su pedido por teléfono al número 91 446 33 50 - Fax 91 445 62 18 - email: clientes@paraninfo.es

**No olvide indicar nº de campaña si el pedido es a través de email**

**Ruego me envíen los títulos que indico a continuación:**

Campaña 155

PARANINFO
THOMSON LEARNING

ISBN: 84-283-2802-1 ☐ **Guía Rápida Excel XP 2002.** 2.500 pts
ISBN: 84-283-2821-8 ☐ **Guía Rápida Word XP 2002.** 2.500 pts.
ISBN: 84-283-2836-6 ☐ **Guía Rápida Access XP 2002.** 2.500 pts.
ISBN: 84-283-2838-2 ☐ **Guía Rápida Windows XP Home Edition 2002.** 2.700 pts.
ISBN: 84-283-2839-0 ☐ **Guía Rápida PowerPoint XP 2002.** 2.700 pts.

Nombre y apellidos ____________________

Calle ____________________ Nº ______

Ciudad ____________________

Distrito Postal __________ Provincia __________

Fecha __________

Teléfono __________ Firma

Para su pago:
☐ Remito giro postal de __________ pts.
☐ Adjunto cheque de __________ pts.
☐ Envíen a reembolso con gastos a mi cargo
☐ Efectúen cargo en tarjeta de crédito:
☐ VISA
☐ AMERICAN EXPRESS

Número __________
Titular __________
Fecha de caducidad __________

Firma del titular de la tarjeta

# Fuente de alimentación digital para banco de trabajo (3)

## parte 3: el programa

Diseñado por R. Pagel

**La fuente de alimentación digital para banco de trabajo está controlada por un microcontrolador programado en el lenguaje PIC BASIC, mientras que un programa escrito en Visual BASIC es el responsable de reproducir y visualizar el panel de control en un PC.**

La **Figura 1** muestra el diagrama de flujo del programa que está almacenado en el microcontrolador. En el inicio del mismo se ejecuta una breve secuencia de inicio, la cual resetea el conjunto de valores y les asigna un valor 0, además de configurar algunos de los terminales del microcontrolador. El siguiente paso, en el que se realizan las medidas de los valores de las tensiones y corrientes actuales, ya forma parte del bucle del programa principal. Todos los elementos restantes del programa siguen de forma secuencial este bucle. Puede darse una desviación cuando se están leyendo los distintos pulsadores. El procedimiento para la lectura de los botones es de hecho tan engorroso (e inevitable) como se muestra en el diagrama de flujo. La idea es leer cada botón por turno y, cuando encontremos que uno de ellos ha sido pulsado, el microcontrolador almacena el correspondiente código de la tecla. Por último, ante la cuestión "button pressed?" (es decir, "¿botón pulsado?"), el microcontrolador verifica si alguno de los botones ha sido pulsado. Si encuentra que sí, tomamos el camino del código que incrementa o decrementa el valor correspondiente, siempre y cuando el valor permanezca dentro del rango permitido. A continuación se sucede un retardo de medio segundo antes de que los pulsadores se verifiquen de nuevo. Esto proporciona una función de repetición automática. Si no se ha pulsado ningún botón, el programa toma el camino de retornar al punto inicial del bucle principal para medir la tensión y la corriente actual de nuevo.

## El programa en BASIC

El listado del código fuente para el microcontrolador aparece en la **Figura 2**. El programa del microcontrolador, escrito en el lenguaje PIC BASIC 1.3, lo podemos bajar de la página web www.pic-basic.de.

El lenguaje PIC BASIC permite que los programas para el microcontrolador puedan escribirse rápida y fácilmente. También realiza la compilación del programa y la programación en el propio circuito integrado de forma sencilla. En la página web www.pic-basic.de podemos encontrar información adicional del lenguaje PIC BASIC, así como la última versión actualizada del programa. En el momento de la escritura de este artículo, la información para el lenguaje PIC BASIC sólo estaba disponible en alemán. Esperamos que Mr Pagel genere prontamente la correspondiente versión traducida al inglés.

La **Figura 3** nos muestra la programación en el propio sistema de la fuente de alimentación de 1 A, utilizando el programador PIC BASIC.

En primer lugar se declaran todas las variables utilizadas en el programa. Existen 13 variables con un ancho de "byte" (8 bits) y dos variables con un ancho de "word" (16 bits), ocupando un total de 17 bytes en la memoria RAM del microcontrolador (además, el programa PIC BASIC ha reservado un total de 12 bytes adicionales en un área específica). A continuación sigue la primera parte del programa: ésta es la parte indicada en el diagrama de flujo por "inicialización". La etiqueta "**Start**" ("Inicio") marca el punto de entrada para el bucle principal. El programa dispone de los suficientes comentarios como para que no sea necesario una descripción detallada en esta parte del artículo. Sin embargo, sí debemos realizar algunas puntualizaciones sobre algunos elementos.

### Conversor A/D

El comando "ADW A2, 5380, 0, Meas_Voltage" nos proporciona una conversión analógica a digital y escribe el valor medido en la variable Meas_Voltage (medida de tensión). El factor de escala puede ajustarse

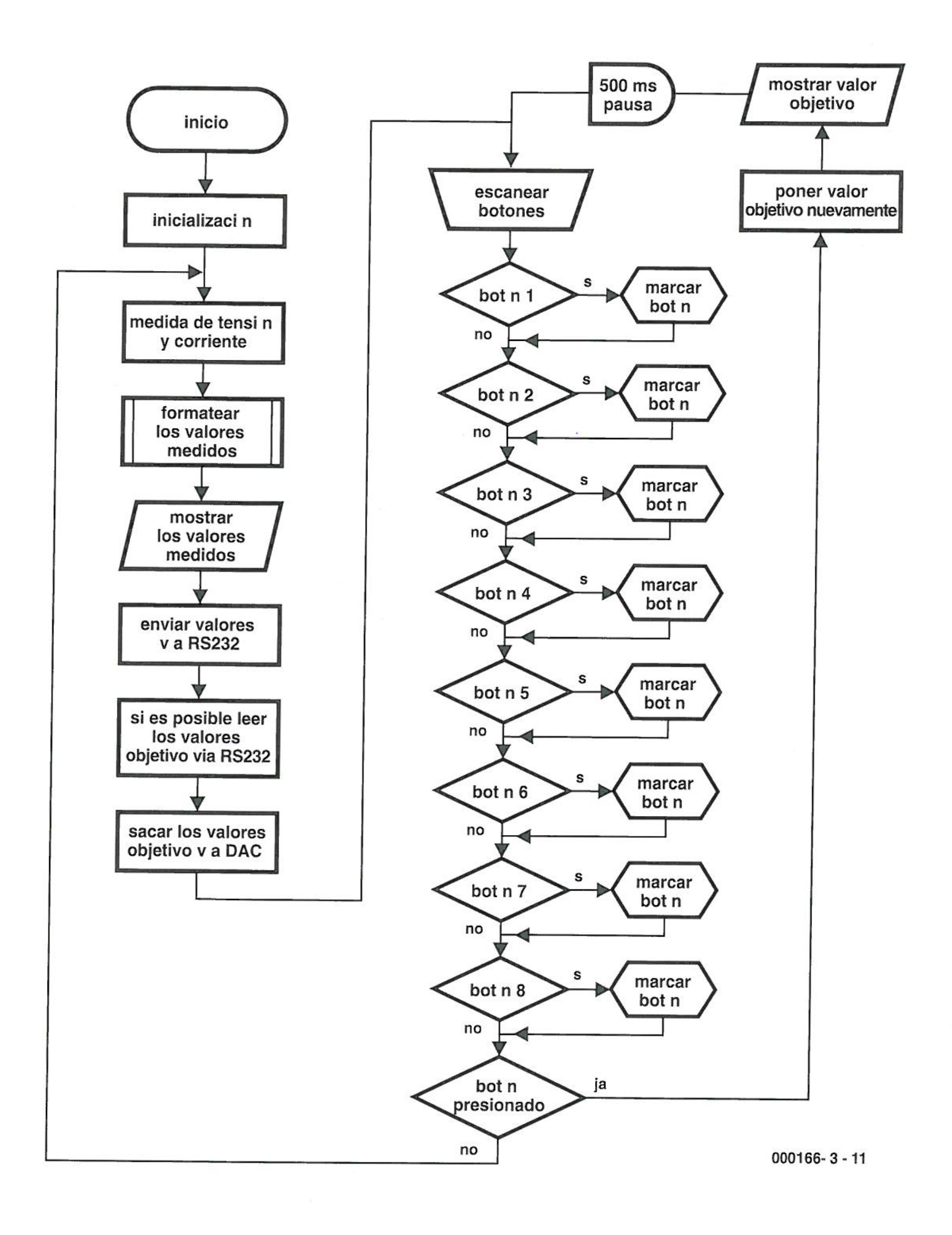



Figura 1. Diagrama de flujo del programa del microcontrolador.

cambiando el valor 5380 que se ha marcado en el comando. Sin embargo, el circuito está diseñado para que este valor no tenga que cambiarse.

**Subrutinas en código ensamblador**

Bajo el nombre "Format measured values" se hace una llamada a la subrutina Format del código ensamblador. Esta subrutina tiene una longitud de 30 bytes y utiliza algunas subrutinas del PIC BASIC, proporcionadas para hacer uso de otros comandos BASIC, que dan formato a los valores leídos para ser presentados en pantalla. Utilizando este truco podemos ahorrar una gran cantidad de la preciada memoria del programa.

La subrutina de ocho bytes "RS232E" también está escrita en ensamblador. Esta subrutina configura la señal CTS a nivel alto y espera durante un cierto tiempo si hay datos que están siendo enviados desde el PC. Si es así, la subrutina de ensamblador llama a la subrutina "Read" del BASIC, que es responsable de la lectura actual del dato de entrada.

El resto de los elementos que quedan del programa están escritos enteramente en BASIC. El tamaño del código compilado para el

```
'D-PSU 25V, 2.5A or 20V, 1A

';Atención: las modificaciones en el programa requieren que los
      números de los registros
'en las subrutinas del ensamblador sean verificadas en los dis-
      tintos cambios!

'_____________________________

'declaración de variables
VarB Lh1, Lh2, Lh3, Lh5, Lh6, Lh7, Uvalue, Ivalue, y
VarB Buttonnumber, Accu, Callcounter, Bitpattern
VarW Meas_Voltage, Meas_Current

'_____________________________

'Programa Principal
Init:
CV Uvalue, Ivalue  'configura a 0 en cada inicio'
Low A3             'salida CA/D a 0
High B2            'CTS: no listo para recibir

Start:
'Medida de la tensión y de la corriente
'El uso del valor 5??? permite ajustar el valor de escala del C A/D
'                                + - 20 igual aprox. 1 dígito
Low A4    'Mux to U
ADW A2, 5380, 0, Meas_Voltage  'Medida de  Tensión
Meas_Voltage = Meas_Voltage Shr 1   'igual / 2
High A4   'Mux to I
ADW A2, 5380, 0, Meas_Current  'Medida de corriente
Meas_Current = Meas_Current Shr 1   ' igual / 2   'línea fuente
      para 2,5A
'Meas_Current = Meas_Current Shr 2   ' igual / 4   'línea
      fuente para 1A

'Formato de los valores medidos
Call Format

'Muestra los valores medidos en el LCD
LCD B5, " ", Lh1, Lh2, ",", Lh3, "V     ", Lh5, ",", Lh6, Lh7,
      "A "

'Envía los valores medidos sobre la RS 232
SerOut B3, 9600, "D", #Meas_Voltage, #Meas_Current, 13

'Permite recibir un nuevo valor de ajuste por RS 232
Call RS232E

'Envía el valor de ajuste sobre el C A/D
PWM A1, Uvalue, 64  'Selecciona la tensión (200 = 20 V)
PWM A0, Ivalue, 64  'Selecciona la corriente (200 = 2 A o 200 =
      1 A)

'Chequea botones

Entry:

Accu = %00010000  'Bit 4 Alto (reset al presionar botón)
CV Callcounter, Buttonnumber
Call ButtonScan
Call ButtonScan
Call ButtonScan
Call ButtonScan
Call ButtonScan
Call ButtonScan
Call ButtonScan
Call ButtonScan
Branch Buttonnumber, Start, Button1, Button2, Button3, Button4,
      Button5, Button6, Button7, Button8

Button1:
If Uvalue > 240 Then Skip    'Línea para fuente 2,5 A
'If Uvalue > 190 Then Skip   'Línea para fuente 1 A
Uvalue = Uvalue + 10
Goto Display_Uvalue

Button2:
If Ivalue > 240 Then Skip    'Línea para fuente 2,5 A
'If Ivalue > 190 Then Skip   'Línea para fuente 1 A
Ivalue = Ivalue + 10
Goto Display_Ivalue

Button3:
If Uvalue < 10 Then Skip
Uvalue = Uvalue - 10
Goto Display_Uvalue

Button4:
If Ivalue < 10 Then Skip
Ivalue = Ivalue - 10
Goto Display_Ivalue

Button5:
If Uvalue > 249 Then Skip    'Línea para fuente 2,5A
'If Uvalue > 199 Then Skip   'Línea para fuente 1A
Inc Uvalue
Goto Display_Uvalue

Button6:
If Ivalue > 249 Then Skip    'Línea para fuente 2.5A
'If Ivalue > 198 Then Skip 2  'Línea para fuente 1A
Inc Ivalue
'Inc Ivalue  'Línea para fuente 1A (omitir para version de 2,5
      A)
Goto Display_Ivalue

Button7:
If Uvalue < 1 Then Skip
Dec Uvalue
Goto Display_Uvalue

Button8:
If Ivalue < 1 Then Skip     'Línea para fuente 2.5A
'If Ivalue < 2 Then Skip 2   'Línea para fuente 1A
Dec Ivalue
'Dec Ivalue  'Línea para fuente 1A (omitir para versión 2,5 A)


Display_Ivalue:
'y = Ivalue Shr 1 'equals / 2  'Línea para 1A (omitir para ver-
      sión 2.5A)
LCD B5, "      ", #Ivalue, "0mA"  'Línea para 2.5A
'LCD B5, "      ", #y, "0mA"  'Línea para 1A
Pause 500
Goto Entry


Display_Uvalue:
LCD B5, "      ", #Uvalue, "00mV"
Pause 500
Goto Entry


'_____________________________

'Subrutinas

'Dependiendo del valor en Callcounter, ButtonScan desplaza uno de los
'ocho bits de la ristra hacia el terminal del HC164.
'Sólo el botón situado en el terminal con el 0 activo
'puede pasar PB4 a nivel Bajo. En ese momento, PB4 indica si se
      ha pulsado o no,
'un botón , mientras que Callcounter proporciona la identidad
      del botón


Sub ButtonScan
   LookUp Callcounter, %11101111, %11011111, %10111111,
      %01111111, %11111011, %11110111, %11111110, %11111101,
      Bitpattern
   EXPo B5, Bitpattern, 0  'sólo Button 0 de la ristra de bits
      puede pasar B4 a nivel Bajo
   Inc Callcounter
   PBI %00010000 = Accu        'sólo lee el bit 4 del puerto B
   If Accu <> 0 then Skip     'salta cuando no se pulsa ningún botón
   Buttonnumber = Callcounter        'marca el número de Botón
EndSub
```

Figura 2. Listado del programa en el PIC BASIC.

```
'La subrutina básica Read se llama desde la
'subrutina ensamblador RS232E

Sub Read
   SerIn B0, 9600, #Uvalue, #Ivalue
   Uvalue = Uvalue Min 250             'límite a 25 Voltios
      'Línea para fuente 2,5 A
   'Uvalue = Uvalue Min 200             'límite a 20 Voltios
      'Línea para fuente 1 A
   Ivalue = Ivalue Min 250             'límite a 2,5 Amperios
      'Línea para fuente 2,5 A
   'Ivalue = Ivalue Min 200             'límite a 1 Amperio
      'Línea para fuente 1 A
   Y = 1          'Abandona el lazo inmediatamente
Endsub


'El formato ensamblador de la subrutina utiliza los recursos ya
      disponibles
'para PB. Carga los registros Lh1-Lh8 con los valores ASCII
      para los
'números 0-9, de acuerdo a los valores que presentan las variables
'variables Meas_Voltage y Meas_Current.

'La subrutina auxiliar llamada Packer salva los 8 bytes de la
      memoria de programa
'Packer llama al programa  en código máquina SOSS°, el cual
      está contenido en la
'salida del compilador PB, cuando se utilizan los comandos
      SerOut - #WordVar
'o LCD - #WordVar".
'Este programa devuelve  el valor decimal equivalente de una
      variable word.
'Divide el valor contenido en HWERT2/R21 por el valor prove-
      niente de la
'tabla de salto SOTT° (también contenida en la salida del com-
      pilador).
'El valor(!) en el FSR tiene que ser el valor ADD
'de la tabla de salto (Pos. 5 = 0, 4 = 2, 3 = 4, 2 = 6, 1 =
      Rest en R21).
'LWERT1 contiene el código ASCII (caracteres 0-9) como resultado.

Ass Format
      ;format voltage
   MOVF 24,W
   MOVWF HWERT2
   MOVF 23,W
   MOVWF 21
   MOVLW 2
   Call Packer
   MOVWF 27
   MOVLW 4
   Call Packer
   MOVWF 28
   MOVLW 6
   Call Packer
   MOVWF 29
      ;format current
   MOVF 26,W
   MOVWF HWERT2
   MOVF 25,W
   MOVWF 21
   MOVLW 2
   Call Packer
   MOVWF 30
   MOVLW 4
   Call Packer
   MOVWF 31
   MOVLW 6
   Call Packer
   MOVWF 32
   Return

Packer:       ;la no repetición de líneas ahorra 8 bytes de
      memoria de programa
   MOVWF FSR
   CALL SOSS°
   MOVF LWERT1,W
EndAss


'RS232E controla la recepción de datos en la interfaz. Cada vez
      que es llamada, la línea CTS se coloca a nivel alto
      durante 1,5 ms
'Si llega un carácter a través de la línea RxD dentro de este
      periodo, el D-PSU pasa a modo recepción, por ejemplo, y
      se llama a la subrutina Read.
'A continuación, tienen que llegar 2 valores, con su CR de fina-
      lización,
'al interfaz, antes de que el controlador tenga permiso para
'abandonar la subrutina

Ass RS232E
   CLRF 35          ;Clrf Y (= R35)
 RS232:
   BCF PB,2         ; CTS: listo para recibir
   BTFSS PB,0       ; RxT terminal de prueba
   Call Read
   DECFSZ 35,F      ;
   GOTO RS232       ; repite el bucle 256 veces
   BSF PB,2         ; CTS: no listo para recibir
EndAss
```

programa completo es de unos 1.009 ó 1.021 bytes (para la versión de 2,5 A y de 1 A, respectivamente), de manera que se puede cargar todo en la memoria del programa del PIC 16F84. Las líneas de programa que tienen que cambiarse para las dos versiones de la fuente de alimentación, están marcadas en el listado del programa en BASIC.

Si deseamos habilitar el temporizador de vigilancia en el microcontrolador, la palabra de configuración en la salida del ensamblador debe ser cambiada como sigue:

```
CONFIG  B'11111111110101'
```

Además, se debe insertar una instrucción "CLRWDT" en un determinado punto del código, en el bucle principal y en el bucle de búsqueda del pulsador activado. El último bucle se ejecuta en tan sólo 500 ms, mientras que el bucle principal se ejecuta en unos 780 ms. Con los valores seleccionados en el registro "Option" (Opciones) en el momento del encendido, el temporizador de vigilancia generará una señal de reset que se producirá 2,3 segundos después de este arranque. Este tiempo es suficiente para que ambos bucles se ejecuten (incluso la mitad de este tiempo podría ser suficiente). El lugar más sencillo donde colocar la instrucción CLRWDT es inmediatamente después de la etiqueta Entry (Entrada). Por lo tanto, el código que sigue se desplaza hacia abajo en una posición, algo que no produce diferencias en este caso.

**El protocolo de la interfaz**

El paquete de datos que la fuente de alimentación envía sobre su interfaz está estructurado de la siguiente manera:

```
Duuuuuiiiii(
```

Primero se envía una "D", seguida por un valor de cinco dígitos para la tensión y otro valor de cinco dígitos para la corriente, terminando el paquete con un carácter de retorno de carro. El dígito menos significativo del valor de la tensión representa un valor de 10 mV, mientras que el dígito menos significativo de la corriente representa un valor de 1 mA. Por lo tanto, los dos valores se entregan con una resolución 10 veces superior a la que se muestra en el panel LCD. El primer dígito de los valores de la tensión y de la corriente siempre es cero.

Cuando queremos enviar los valores de

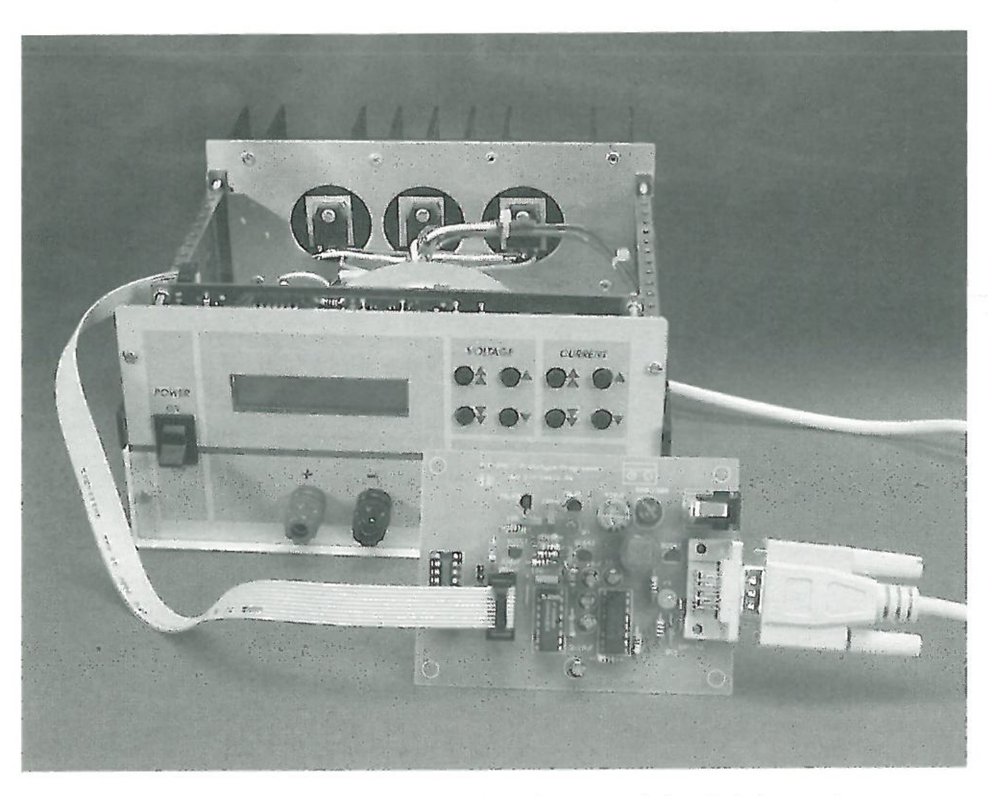



Figura 3. Programación en el propio sistema del microcontrolador de la fuente de alimentación.

configuración hacia la fuente de alimentación, debemos enviar tanto los valores de configuración de la tensión como los de la corriente, uno inmediatamente después del otro. En ambos casos, se pueden enviar hasta tres dígitos:

```
uuu( iii(
```

Después de cada valor se debe de enviar un carácter no digital (por ejemplo el carácter de retorno de carro). Para cualquiera de los dos modelos de fuente de alimentación, si queremos seleccionar un valor de 20 V en la fuente, dicho valor se debe enviar como "200". En el caso de la corriente, si deseamos seleccionar 1 A de corriente se deberá enviar un valor de "100" para el caso de la fuente de 2,5 A, o de "200" para el modelo de fuente de 1 A.

## El programa en Visual BASIC

El programa de control, escrito especialmente para este proyecto, se ejecuta bajo Windows 95 ó 98. Su interfaz de control se asemeja al panel frontal de una fuente de alimentación (ver **Figura 4**). Al igual que sucede en la realidad, los valores de configuración de la tensión y de la corriente pueden ajustarse utilizando ocho botones de control. Cuando el botón del ratón se suelta es el momento en que los valores se envían hacia la fuente de alimentación. Tanto los valores seleccionados como los actuales se muestran en el panel LCD simultáneo. Si hacemos "click" en una zona de la ventana del programa principal que no sea la de los botones, aparece una ventana de configuración. En dicha ventana podremos configurar la versión de la fuente de alimentación (2,5 ó 1 A) y la del puerto serie COM (1 a 4). Si tenemos un ratón conectado al puerto COM1 es posible que el programa no funcione correctamente con la fuente de alimentación conectada al COM3, pero esto es un problema común en todos los ordenadores.

En el lado derecho de la ventana de configuración podemos especificar los nombres de un fichero lógico y de un fichero de control. Haciendo "click" con el ratón en el botón "Start" ("Inicio") asociado, el fichero correspondiente podrá ser leído o escrito: el fichero contiene las lecturas mostradas, además de un tiempo adicional que nos indica cuándo han cambiado dichas lecturas (en el caso del fichero lógico), o cuándo tienen que cambiar (en el caso del fichero de control ya preparado). En el caso más sencillo, el fichero de control puede ser creado a partir del fichero lógico, modificando el tiempo de registro de la medida.

El siguiente ejemplo de datos grabados, muestra el formato utilizado por los ficheros de control y lógico:

```
#2000 - 08 - 20
14:35:53#, "04.9V", "0.97ª"
```

Entre los dos símbolos de almohadilla (#) tenemos el dato de la fecha y la hora en la que se ha producido el cambio indicado o en la que tiene que ocurrir. Entre el tiempo marcado y los dos valores eléctricos de tensión y de corriente, se coloca un carácter de separación, que en nuestro caso es una ",". Esto permite, por ejemplo, que se pueda realizar un procesamiento y una presentación gráfica del contenido del fichero utilizando un programa de hojas de cálculo.

(000100-3)

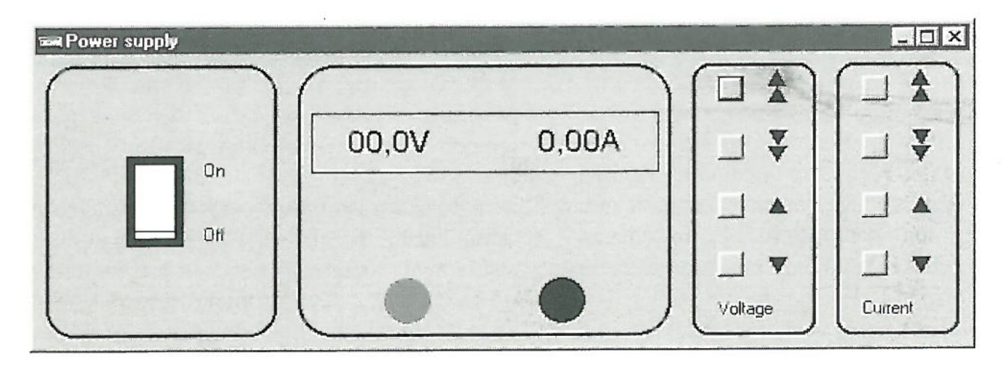


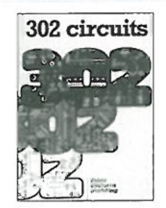

Figura 4. Visualización de la fuente de alimentación en el monitor del PC.

*Elektor ofrece a sus lectores una selección de libros técnicos de gran valor para aficionados y profesionales. Todos ellos están escritos en inglés, el idioma más utilizado en el ámbito técnico.*

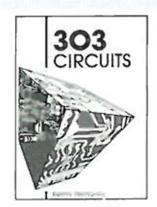
## SERIE 300. Circuitos y aplicaciones:



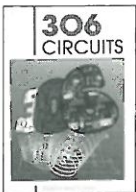

### 301 Circuits
*Precio: 27,05 €*



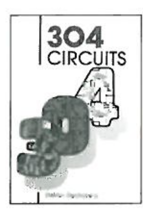

### 302 Circuits
*Precio: 27,05 €*



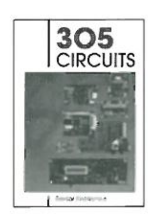

### 303 Circuits
*Precio: 29,45 €*



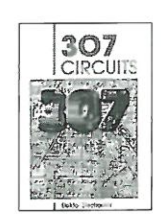

### 306 Circuits
*Precio: 42,07 €*




### 304 Circuits
*Precio: 34,26 €*




### 305 Circuits
*Precio: 39,67 €*

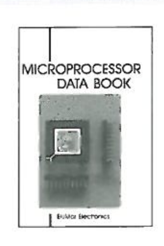



### 307 Circuits
*Precio: 46,28 €*

## Microprocesadores, periféricos y técnicas:



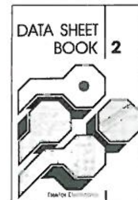

### Short course 8051/8032 microcontrollers and assemble
*Precio: 27,05 €*



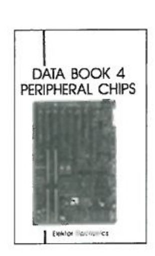



### Microprocessor Data Book
*Precio: 29,45 €*

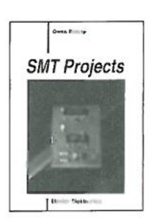
### Data Book 3 - Peripherical Chips
*Precio: 29,45 €*




### Data Sheet Book 2
*Precio: 27,05 €*




### Data Book 4 - Peripherical Chips
*Precio: 29,45 €*




### SMT Projects
*Precio: 27,05 €*



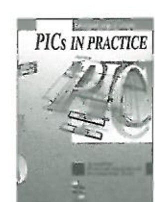

### ST62 Microcontrollers
*Precio: 66,11 €*




### SCSI The ins and outs
*Precio: 66,11 €*




### The I²C Bus
*Precio: 66,11 €*

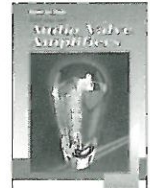



### PICs in practice
*Precio: 66,11 €*

## Construcción de equipos completos:



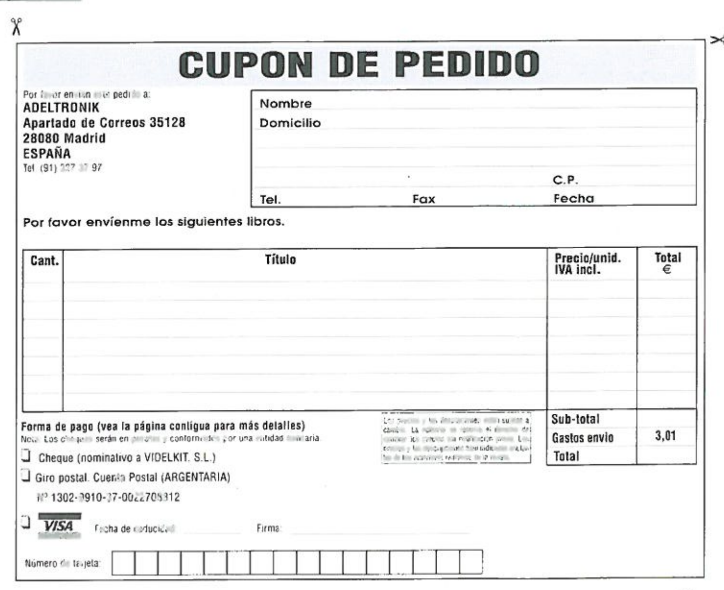

### Build your own Electronic test instruments
*Precio: 42,07 €*




### Build your own Audio Valve Amplifiers
*Precio: 39,67 €*




### Build your own High-end Audio Equipment
*Precio: 39,67 €*



## CUPON DE PEDIDO

Por favor envíen este pedido a:
**ADELTRONIK**
**Apartado de Correos 35128**
**28080 Madrid**
**ESPAÑA**
Tel. (91) 327 37 97

Nombre
Domicilio
C.P.
Tel. Fax Fecha

Por favor envíenme los siguientes libros.

| Cant. | Título | Precio/unid. IVA incl. | Total € |
|---|---|---|---|
| | | | |
| | | Sub-total | |
| | | Gastos envío | 3,01 |
| | | Total | |

**Forma de pago (vea la página contigua para más detalles)**
Nota: Los cheques serán en pesetas y conformidad por una entidad bancaria.

- ❑ Cheque (nominativo a VIDELKIT. S.L.)
- ❑ Giro postal. Cuenta Postal (ARGENTARIA) Nº 1302-0910-37-0022705312
- ❑ VISA Fecha de caducidad Firma

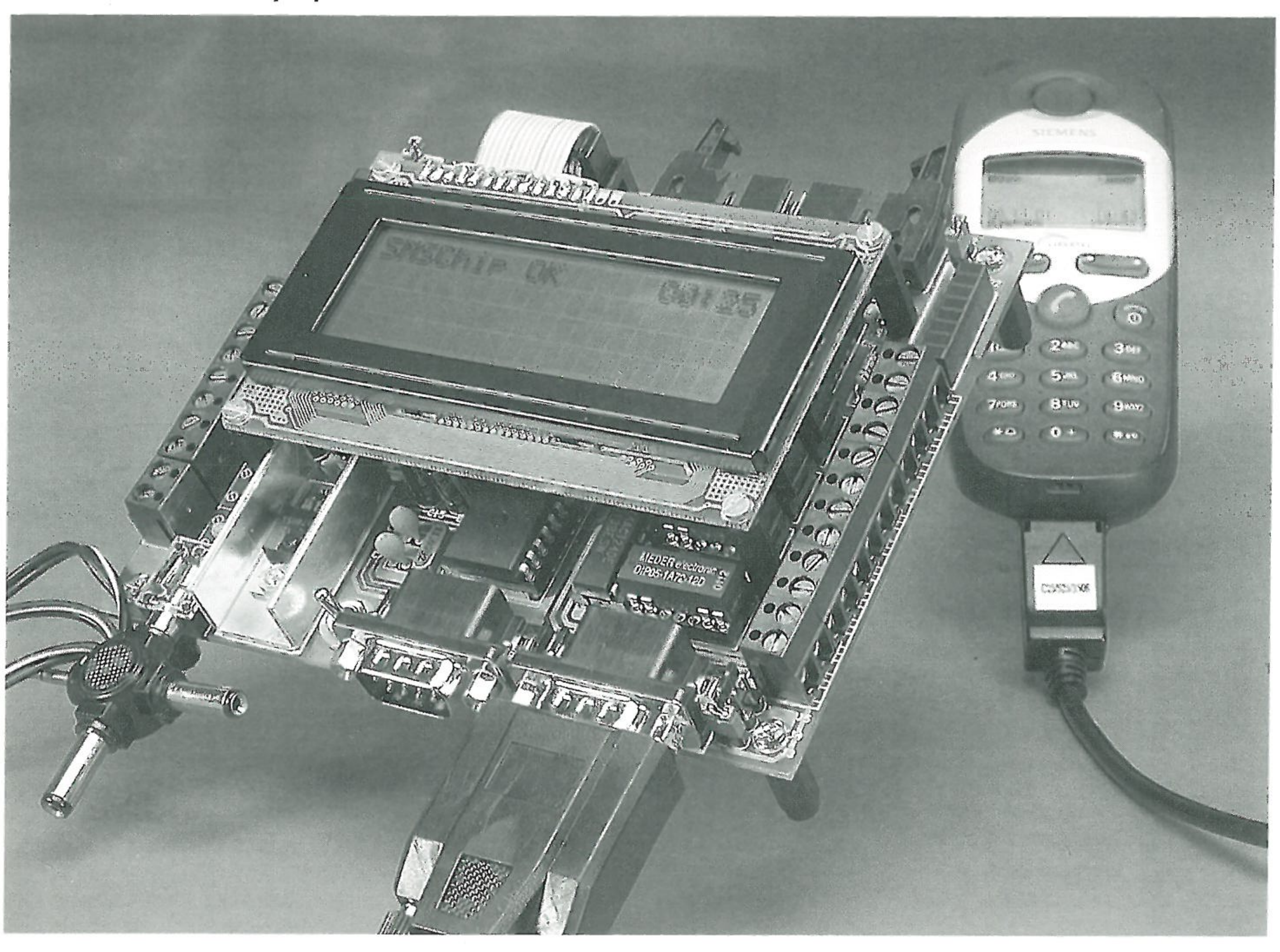
Número de tarjeta:

# Control Remoto de Procesos mediante un Teléfono Móvil

## Para la serie "35" de teléfonos móviles de Siemens

Diseñado por Prof. B. vom Berg and P. Groppe (Georg Agricola Technical University)

El Sistema de Mensajes Cortos (SMS) de un teléfono móvil estándar puede emplearse para muchas más aplicaciones que las de intercambiar mensajes escritos. El montaje de esta aplicación nos permite hacer trabajar al teléfono móvil en una localización remota, verificando y controlando equipos externos.



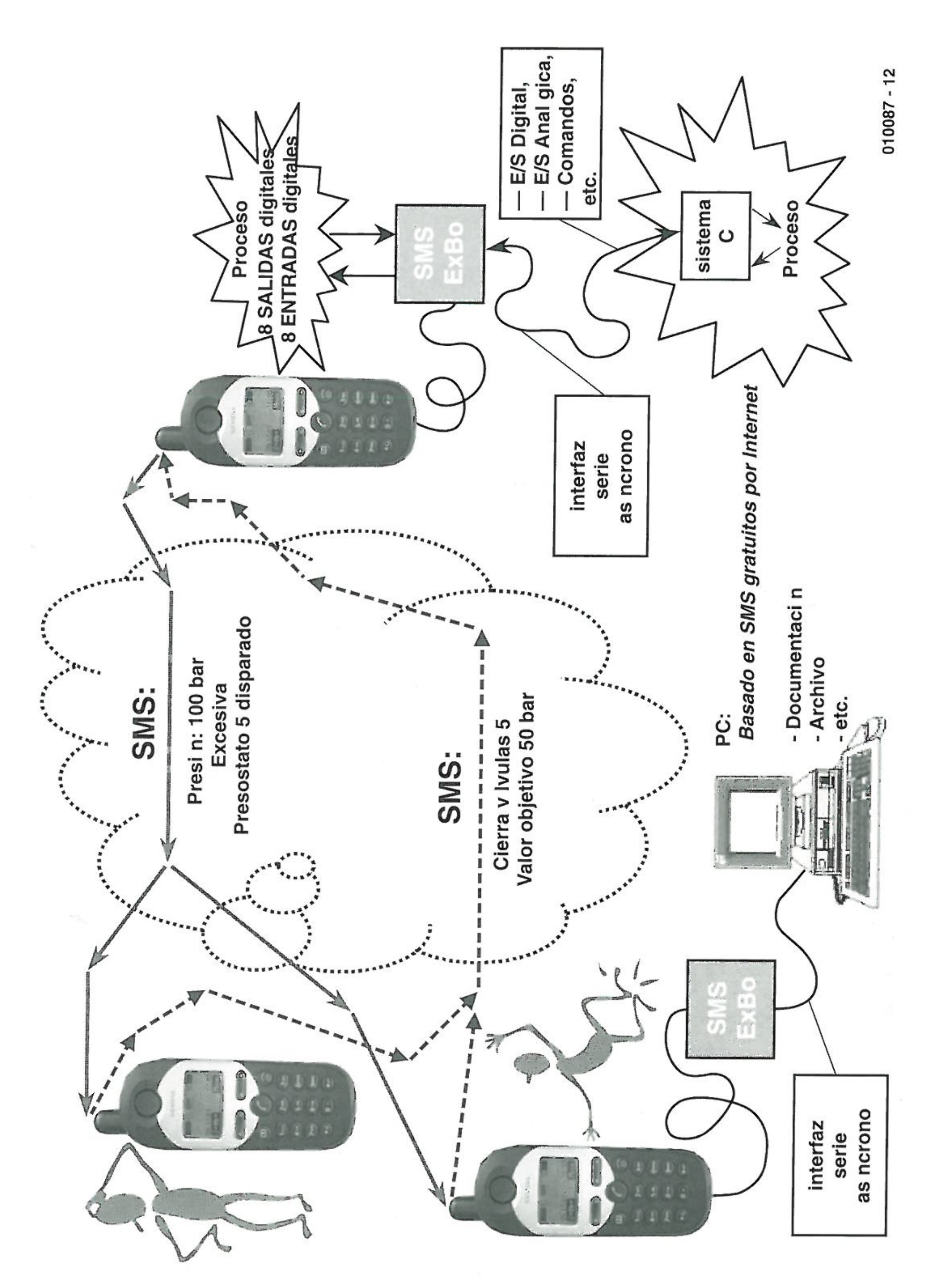



Figura 1. Control remoto de procesos utilizando un móvil y el servicio SMS.

Este diseño es el resultado de la colaboración entre una universidad técnica (TFH Georg Agricola) y la compañía Engelmann & Schrader. Juntos han producido un controlador remoto de procesos profesional y flexible en la forma de una placa de experimentación (la TFH SMSD ExBo). Esta tarjeta se conecta entre los procesos que tienen que ser controlados y un teléfono móvil, permitiéndonos que los procesos puedan ser controlados y monitorizados de forma remota utilizando los mensajes SMS.

## El circuito integrado SMS

La placa interfaz SMS ofrece dos niveles de sofistificación en el control. En el nivel básico, la tarjeta interpreta los mensajes SMS, verifica la contraseña de entrada y controla directamente los relés de salida o los indicadores.

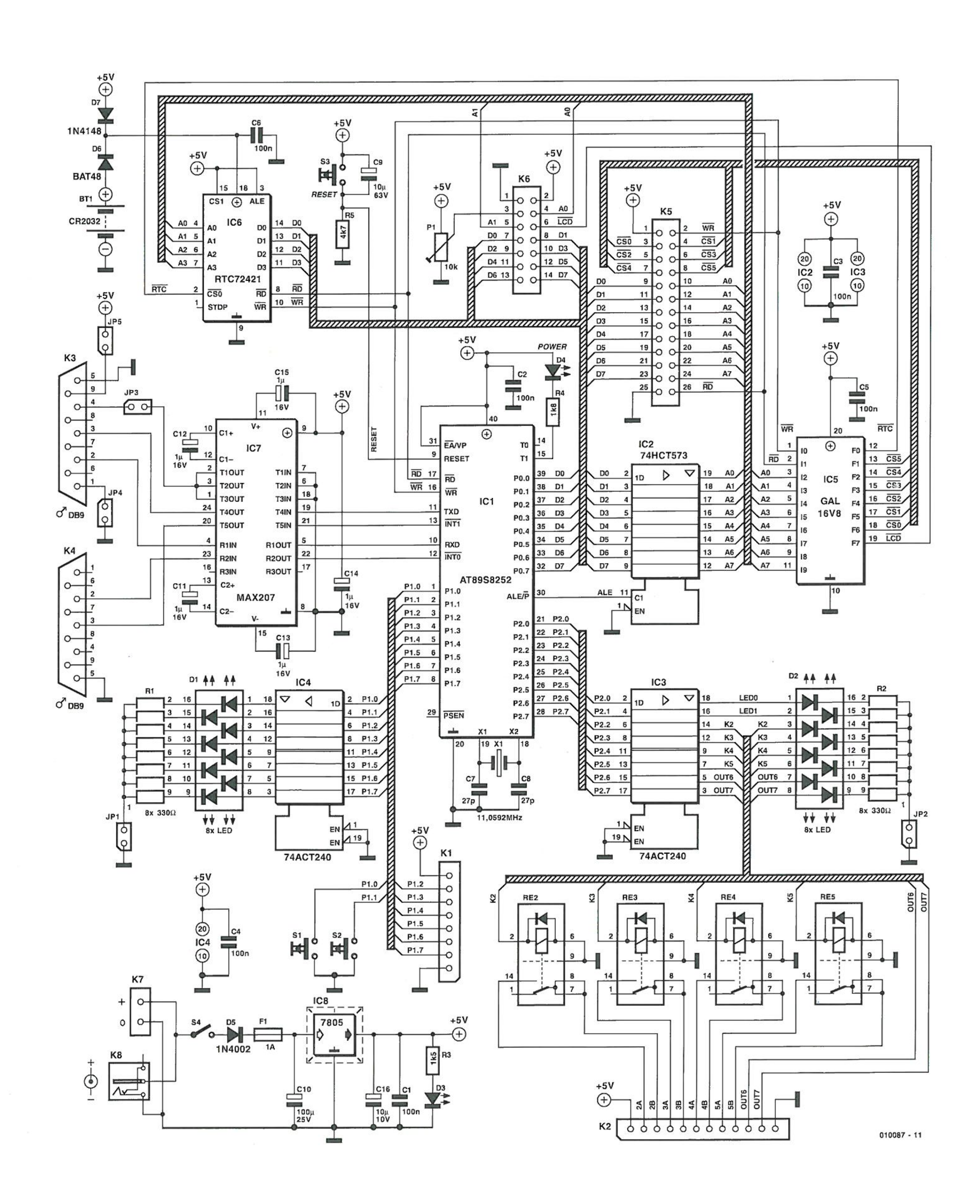



Figura 2. Esquema eléctrico del circuito.

| Tabla 1. Especificaciones de E/S del puerto | | |
|---|---|---|
| **Puerto** | **Nº Terminal** | **Función** |
| P1.0 | 0 | ENTRADA: conexión para el conmutador S1 |
| P1.1 | 1 | ENTRADA: conexión para el conmutador S2 |
| P1.2 | 2 | ENTRADA: nivel TTL, no protegido. |
| P1.3 | 3 | ENTRADA: nivel TTL, no protegido. |
| P1.4 | 4 | ENTRADA: nivel TTL, no protegido. |
| P1.5 | 5 | ENTRADA: nivel TTL, no protegido. |
| P1.6 | 6 | ENTRADA: nivel TTL, no protegido. |
| P1.7 | 7 | ENTRADA: nivel TTL, no protegido. |
| P2.0 | 8 | SALIDA: conexión LED0 del grupo de LEDs D2. |
| P2.1 | 9 | SALIDA: conexión LED1 del grupo de LEDs D2. |
| P2.2 | 10 | SALIDA: relé Re2; máx. 200 VDC, máx. 1 A, máx. 15 W. |
| P2.3 | 11 | SALIDA: relé Re3; máx. 200 VDC, máx. 1 A, máx. 15 W. |
| P2.4 | 12 | SALIDA: relé Re4; máx. 200 VDC, máx. 1 A, máx. 15 W. |
| P2.5 | 13 | SALIDA: relé Re5; máx. 200 VDC, máx. 1 A, máx. 15 W. |
| P2.6 | 14 | SALIDA: nivel TTL del circuito integrado controlador 74AC/HCT240. |
| P2.7 | 15 | SALIDA: nivel TTL del circuito integrado controlador 74AC/HCT240. |

En la otra dirección, el circuito integrado SMS de la tarjeta interfaz inserta información en un mensaje SMS que, posteriormente, es enviado sobre el conector de la interfaz serie hacia el teléfono móvil desde donde será enviado a cualquier teléfono móvil SMS que esté a la escucha en cualquier parte del mundo. La información de control en el mensaje SMS podría ser, por ejemplo, configurar 10 (set 10), lo que tendría un efecto de seleccionar la salida 10 en la tarjeta interfaz y conmutarla sobre cualquier diodo LED, motor o relé conectado a esta salida. De forma similar, en la otra dirección, el circuito integrado SMS lee las entradas de la tarjeta interfaz, provenientes del proceso controlado, y genera el mensaje SMS adecuado que será pasado sobre la interfaz serie hacia el teléfono móvil remoto, donde dicho mensaje será enviado a cualquier teléfono móvil SMS disponible en cualquier parte del mundo. El mensaje puede llevar información como "Presión excesiva detectada en el tanque 3" o "Alerta de intruso en la puerta 5" o, incluso, "Caldera de calefacción de casa de campo encendida". Cómo podemos ver, las posibilidades son infinitas.

La placa interfaz TFH SMS ExBo (Experimental Board, es decir Placa Experimental), está equipada con un segundo puerto interfaz serie asíncrono, lo que permite la conexión de un ordenador externo o un controlador de procesos en el punto remoto. Esta facilidad proporciona un mayor nivel de control y de monitorización, lo que permite que los valores medidos, las variables y la información del estado del proceso se pueda intercambiar sin mayores problemas. Con esta configuración, la placa interfaz iniciará el teléfono móvil como en el caso anterior, pero con la diferencia de que dejará pasar el mensaje SMS hacia el ordenador externo sin llegar a alterarlo. Esto proporciona al sistema una mayor flexibilidad con los mensajes SMS, ya que serán codificados y generados en el ordenador externo.

## El circuito que acompaña al integrado SMS

El microcontrolador utilizado en este diseño es el AT 89S8252 o el AT 89LS8250 de la casa Atmel. Este microcontrolador está basado en el procesador 8051 y dispone de una memoria de programa Flash de 8 Kbytes junto con una memoria Flash de datos de 2 Kbytes. Está programado para actuar como un circuito integrado SMS y, esencialmente, realiza tres tareas básicas:

- Re-programa el teléfono móvil y gestiona las comunicaciones con el teléfono móvil sobre la interfaz serie.
- Recibe y descodifica los mensajes SMSD: vigila y controla las entradas y salidas de la placa. Genera mensajes SMS.
- Pasa mensajes hacia y desde el teléfono móvil hasta el sistema del ordenador externo.

Este circuito integrado dispone de dos puertos de ocho bits (puerto 1 y puerto 2), lo que proporciona un total de 16 líneas de entrada/salida digitales totalmente programables. El circuito integrado también dispone de un Receptor/Transmisor Universal Asíncrono (UART) que gestiona las comunicaciones serie y que, en esta aplicación, se encarga de pasar los datos serie, incluyendo los mensajes SMS, hacia y desde el teléfono móvil. En el programa se ha implementado un segundo interfaz serie para comunicar con el ordenador externo o con el controlador de procesos, a una velocidad de 9.600 baudios.

La placa interfaz también contiene un circuito integrado que proporciona la función de reloj de tiempo real, con una batería de seguridad y un conector para montar una matriz LCD de puntos donde se visualizan los mensajes SMS.

Los comandos serie se envían hacia el móvil utilizando las secuencias de un módem Hayes estándar (comandos AT). Para obtener más información acerca del control de un módem Hayes podemos ver el apartado correspondiente que acompaña este artículo.

## Los circuitos que acompañan al integrado SMS

El circuito integrado SMS, junto con algunos componentes periféricos adicionales, se encarga de generar un mensaje SMS muy básico que gestionará el diseño. Pero si miramos con más detenimiento el esquema eléctrico de la **Figura 2**, podemos ver que la placa interfaz TFH SMS ExBo ha sido diseñada teniendo como base una plataforma muy flexible para el desarrollo de mensajes SMS, de manera que nos permita muchas posibilidades de conexión. El circuito integrado (DIL) SMS IC1 está sincronizado por una configuración estándar con oscilador de cristal (X1, C7 y C8). Además, el montaje está equipado con

## "Chateando" con el móvil

El circuito integrado SMS, utilizado en esta placa interfaz, se comunica con el teléfono móvil utilizando la secuencia de comandos compatible Hayes sobre su puerto de conexión de su interfaz serie. Allá por los años 70 los módems eran unos curiosos periféricos de ordenadores que se quedaban emitiendo sonidos, esperando a que Internet fuese inventada. Existían muchos módems en el mercado con diferentes modelos que ofrecían prestaciones similares, pero con comandos de control incompatibles. La empresa estadounidense Hayes, pionera en este campo, ofreció un conjunto de comandos que podían ser utilizados para controlar el módem,de manera que pronto se convirtió en un estándar y rápidamente se implantó el que todos los módems fuesen compatibles "Hayes". Incluso hoy día esta secuencia de comandos está siendo implementada en todos los módems y teléfonos móviles.

Los comandos se inician normalmente con los caracteres ASCII "AT", pero los comandos actuales que se han convertido en estándar se inician con la secuencia de caracteres "AT +C", acabando todos ellos con el código ASCII de retorno de carro. Estos comandos son conocidos también con el nombre de comandos "AT" o "AT +C". Estos comandos han sido adoptados también recientemente por la industria de la telefonía móvil y han sido definidos en las secciones GSM07.07 y GSM07.05 de la especificación de telefonía móvil GSM, para el control de los teléfonos sobre su interfaz serie. La interfaz puede utilizar niveles de señales V24 a través de un enlace por cable o enlace por infrarrojos IrDa. Todo el conjunto supone un total de 55 comandos AT, los cuales deben de poder ser entendidos por todos los teléfonos móviles GSM actuales. Esto permite que, por ejemplo, se pueda acceder a la agenda del teléfono móvil, gestionar mensajes SMS, ajustar los tonos de llamada y el volumen del altavoz, etc.

La conexión de un cable de enlace de datos (cable de datos) o el uso de un enlace por infrarrojos entre el teléfono móvil y el puerto serie de un ordenador, significa que es posible controlar el teléfono móvil desde el teclado de un ordenador mucho mejor que desde el teclado del propio teléfono. Esto es mucho más fácil para nuestros dedos cuando escribimos mensajes de texto.

Además de los comandos estándar, cada fabricante ha definido un conjunto de comandos extra que sólo serán entendidos por sus propios teléfonos móviles. Con la serie "35" de Siemens (S35i, C35i, M35i) existe hasta un total de 25 comandos adicionales, todos ellos prefijados con el comando AT^S. Esta gran diversidad crea problemas para cualquiera que considere la posibilidad de construir una interfaz SMS universal. Así, es necesario estudiar muy detenidamente las especificaciones del teléfono móvil para garantizar este logro. En este diseño que presentamos, todos estos comandos han sido programados previamente en el circuito integrado SMS, de manera que puede ser conectado directamente a los teléfonos móviles de la serie "35" de la casa Siemens. Para conseguir desarrollar una aplicación de control remoto es necesario trabajar duro sobre la programación del proyecto. Además se ha proporcionado una segunda interfaz en el punto remoto, de manera que se pueda conectar un ordenador externo para proporcionar unas posibilidades de control más complejas. Por supuesto, sólo es necesario especificar este tipo de teléfono en el punto remoto y la comunicación se realizará sobre el aire con cualquier teléfono móvil SMS, en cualquier parte del mundo.

un circuito de reset formado por la resistencia R5, el pulsador S3 y el condensador C9. A través de la salida presente en el terminal P3.5 se puede controlar un diodo LED (D4) de bajo consumo, para indicar que el teléfono móvil GSM está listo para ser utilizado. La **Tabla 1** nos muestra la asignación de terminales de E/S.

### Entradas digitales (P1.2 – P1.7)

El conector formado por el bloque K1 nos permite la conexión de hasta seis señales de entrada. Todas estas entradas están conectadas directamente con el puerto P1 del microcontrolador. El estado de estas señales se muestra en el conjunto de diodos LEDs, D1, a través del "buffer" IC4. Si nuestra aplicación no necesita esta prestación o deseamos mantener el consumo de corriente lo más bajo posible, no tenemos que montar el puente J1 y dejaremos de montar los diodos LED.

Las dos entradas restantes en P1.0 y P1.1 se usan para los conmutadores S1 y S2.

**Tabla 2. Asignación de puentes**

| | | |
|---|---|---|
| J1 | Montar este puente para activar el grupo de diodos LEDs D1, (Muestra el estado de entrada) | |
| J2 | Montar este puente para activar el grupo de diodos LEDs D2, (Muestra el estado de entrada) | |
| J3 | Tensión positiva de alimentación para el cable enlace de datos. | |
| | Este puente se debe montar cuando se utiliza un cable de Enlace de Datos sin bobina, en caso contrario no se debe montar este puente. | |
| J4 | Controla la corriente de carga de la batería en el móvil (no usado en el Siemens S35). | |
| | Montado (Nivel Bajo): | Carga estándar con 5 V a 150 mA. |
| | No Montado (Alta impedancia): | Carga rápida con 5 V a 400 mA. Solo montar si se está usando un cable de enlace de datos específico, con función de carga. (Ver parte 2 de este artículo). |
| J5 | Potencial positivo de carga para la batería del móvil (no usado en el Siemens S35). | |
| | Montado: | Permite que la batería del móvil se pueda cargar cuando se utiliza un cable de enlace de datos específico (ver parte 2 de este artículo). |
| | No Montado: | En todos los otros casos. |

### Salidas digitales (P2.0 – P2.7)

El circuito integrado IC3 amplifica las salidas P2.0 a P2.7. Cuatro de estas salidas se usan para controlar los relés Re2 a Re5 y estos a su vez proporcionan cuatro salidas conmutadas sobre el conector K2. Dos salidas con niveles TTL, OUT6 y OUT7, también están disponibles en este conector. El estado de estas salidas se puede visualizar a través del conjunto de diodos LEDs D2 y, de nuevo, si esta prestación no es necesaria para nuestro montaje, no tendremos que montar el puente J2.

### Las interfaces serie

Existen dos interfaces serie en la placa interfaz. El conector K3 une una de estas interfaces con el teléfono móvil, mientras que K4 conecta la otra interfaz al puerto serie de un ordenador exterior.

El circuito integrado IC7 (MAX 207) convierte los niveles de tensión de la señales de ambas interfaces (V24) en niveles TTL utilizados en la tarjeta interfaz. También se produce una tensión de, aproximadamente, 10 V por las salidas T1OUT, T2OUT y T3OUT para el cable de datos. Al retirar el puente J3 desconectaremos estas tensiones del interfaz (ver Tabla 2). La corriente para cargar la batería del móvil se suministra a partir del terminal 9 del conector K3, siempre y cuando el puente J5 esté montado. El conector Sub-D K4 es utilizado para conectar la placa a un ordenador externo o a un controlador de procesos, una vez que los mensajes SMS puedan ser leídos y recibidos.

### Los circuitos periféricos al integrado SMS

El circuito integrado SMS, al igual que otros circuitos derivados del procesador 8051, requieren algo de circuitería periférica externa. En primer lugar, el bus de datos y de direcciones necesitan ser demultiplexados en el puerto P0. Esto se consigue a través de un biestable tipo D (IC2) utilizando la señal ALE. Estas líneas de direcciones son entonces descodificadas por el circuito integrado GAL IC5, para generar las señales de selección de circuito (chip select), para el resto de los componentes de la placa interfaz. En la tira de terminales que forman el conector K5 disponemos de las líneas de direcciones, además de la señal RD/WR, la señal de selección de circuito (CS) y las líneas de la tensión de alimentación.

El circuito también incluye un reloj de tiempo real y un conector para un módulo LCD y ambos, a su vez, son controlados por el circuito integrado SMS.

### El RTC (Reloj de Tiempo Real)

Para algunas aplicaciones es esencial disponer de una referencia de tiempo suficientemente precisa, para lo cual se ha incluido en el circuito un reloj de tiempo real, IC6. Este circuito integrado mantiene el tiempo correcto del día para el sistema completo TFH SMS ExBo. Para mantener "viva" toda esta información una batería (BT1) se carga a través del diodo D7 y asegura que el circuito integrado RTC no pierde la información de tiempo si falla la tensión de alimentación principal, proveniente del conector K8.

### El LCD alfanumérico

A través del conector K6 se puede conectar un módulo LCD. La imagen en el principio de este artículo muestra un visualizador de cuatro líneas por 20 caracteres, pero la mayoría de los visualizadores alfanuméricos pueden sustituirse, siempre y cuando sean compatibles con el controlador HD 44780 de la casa Hitachi. Al pulsar P1 se permite el ajuste del contraste del visualizador.

### El circuito integrado GAL

Las señales de selección de circuito (Chip Select) en la placa son generadas por IC5, un circuito integrado 16 V8 que es un Grupo de Puertas Lógicas (Gate Array Logic, GAL). El circuito GAL sencillamente lee las direcciones en sus entradas y genera las señales de selección de integrado para los circuitos integrados periféricos. Las seis señales de selección de circuito, CS0 a CS5, también están disponibles en la tira de terminales del conector K5.

### La fuente de alimentación

IC8 es un regulador de tensión fija que proporciona la tensión de + 5 V para la placa interfaz completa y la corriente de carga para el teléfono móvil. La unidad adaptadora de tensión de red debe proporcionar una tensión comprendida en el rango de los 9 a los 12 V, con una corriente de 800 mA (incluyendo la corriente de carga para el móvil), y está conectada a los conectores K7 o K8. El diodo D5 protege el circuito frente a inversiones accidentales de los terminales de entrada de alimentación y el diodo LED D3 es el indicador de encendido. Por su parte, el conmutador S4 proporciona la función de encendido/apagado.

### Los puentes de configuración

La placa de circuito impresa acabada dispone de varios puentes de configuración. El propósito de estos puentes se muestra en la **Tabla 2**.

(010057-1)

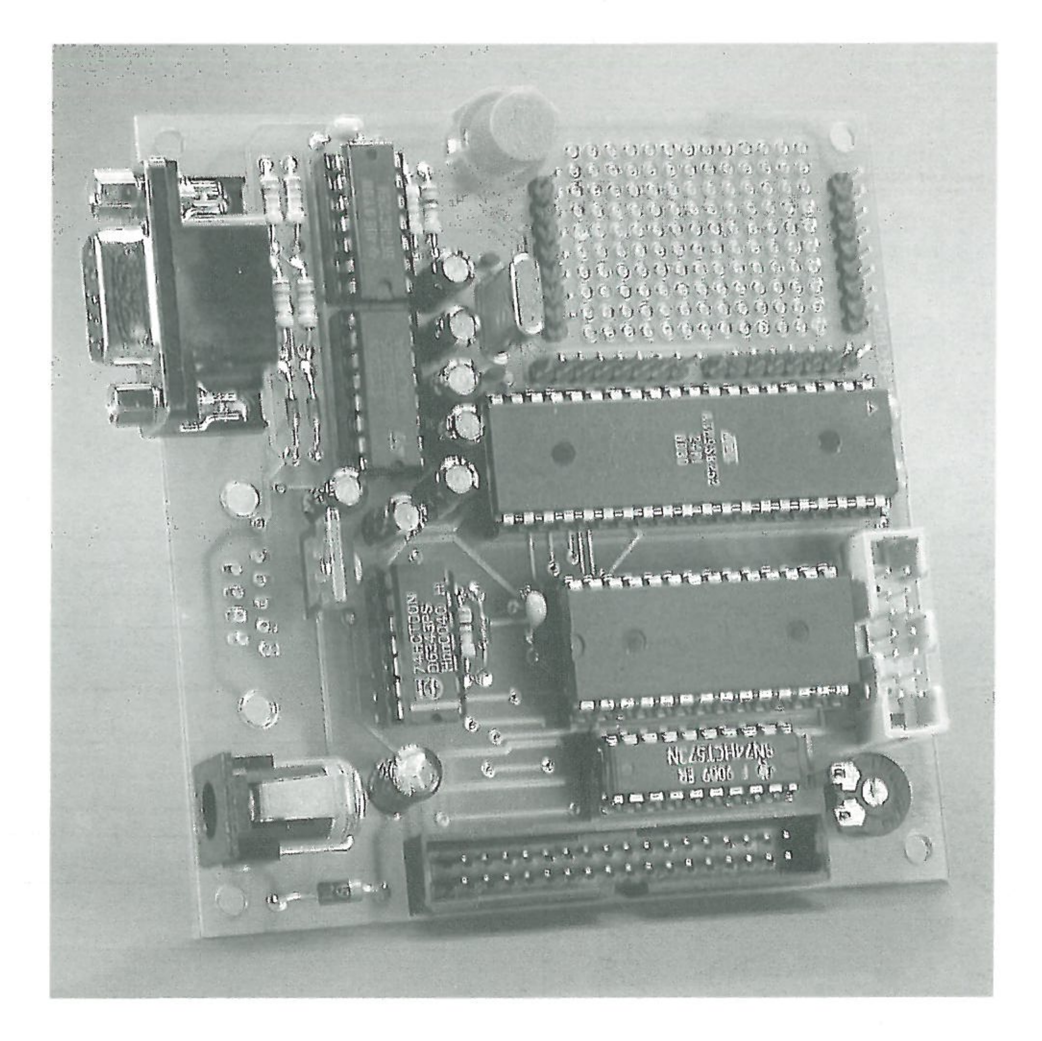
*En la segunda parte de este artículo veremos, de manera más cercana, las conexiones entre el teléfono móvil y la placa interfaz, la configuración básica y la secuencia de comandos para el circuito integrado SMS. También nos detendremos en el análisis de circuito.*

# Curso Básico de Microcontrolador

## parte 1: el ensamblador TASM

Por B. Kainka

**Este curso está dirigido a todo el que quiera saber cómo trabajar con microcontroladores y cómo usarlos, aunque lógicamente no lo abarcaremos todo. Trataremos de explicar los fundamentos y principios desde la base, usando como plataforma hardware la placa de microcontrolador flash 89S8252 publicada recientemente en esta revista.**



En la actualidad, mucha gente pasa gran parte del día trabajando con ordenadores, ya que suponen una gran potencia de trabajo. El corazón de un ordenador es el procesador, concretamente hablamos de un Pentium III.

Comparativamente un microcontrolador es mucho menos y mucho más que el típico procesador de PC. Es menos porque procesa programas pequeños, tiene menos memoria y además es mucho menor. Sin embargo, es más porque ya tiene muchos elementos incorporados en el propio chip, que en otro caso irían dentro de la placa madre de un PC: memoria, temporizadores, interfaces y puertos de conexiones. Esto hace atractivos a los microcontroladores, ya que un programa complejo se puede ejecutar en un simple circuito integrado. Utilizando sólo un programador, alguien puede producir un circuito integrado especial que haga exactamente lo que él quiera y a un coste relativamente bajo.

Figura 1. La Placa Flash 89S8252, que se utiliza en este curso, es un sistema de microcontrolador de propósito general.

Un microcontrolador es algo así como un circuito lógico con muchas entradas y salidas posibles. Lo que hace este circuito está determinado por un programa. Quizás queramos construir un ordenador digital o un reloj despertador, o crear una puerta lógica especial y un generador de reloj universal. Es posible que necesitemos decodificar un sistema digital completo o una señal de control de un circuito digital. En todos estos casos puede ser útil un microcontrolador. Hay muchos ejemplos de problemas cuyas soluciones requieren una enorme placa llena de circuitos integrados y que pueden solucionarse elegantemente mediante un simple circuito integrado llamado microcontrolador. Por ello, será necesario que tengamos algún conocimiento de programación. Hay muchas aplicaciones diferentes que se pueden resolver mediante este dispositivo.

El hardware básico para este curso es la placa del Microcontrolador Flash 89S8252 descrito en un artículo del último mes en Elektor (ver Figura 1). Tal y como ya anunciamos, para la programación del software usaremos los siguientes lenguajes de programación: ensamblador, Basic y C. Nuestros primeros experimentos los realizaremos en ensamblador. ¿Utilizaremos algún ensamblador en particular? ¿Es muy difícil para principiantes? La respuesta es no, porque los ejemplos iniciales son muy pequeños y fáciles de entender. La ventaja de nuestro ensamblador es que permite trabajar muy cerca del hardware, por lo que podremos ver qué es lo que sucede exactamente. Los lenguajes de alto nivel (tales como BASIC), por el contrario, ocultan mucho de lo que ocurre realmente.

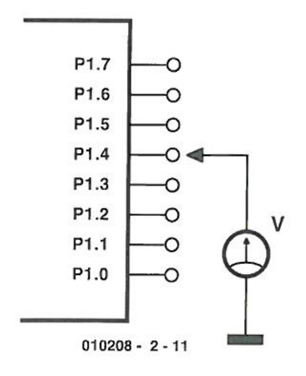



Figura 2. Utilizando un voltímetro se pueden comprobar los resultados del primer experimento.

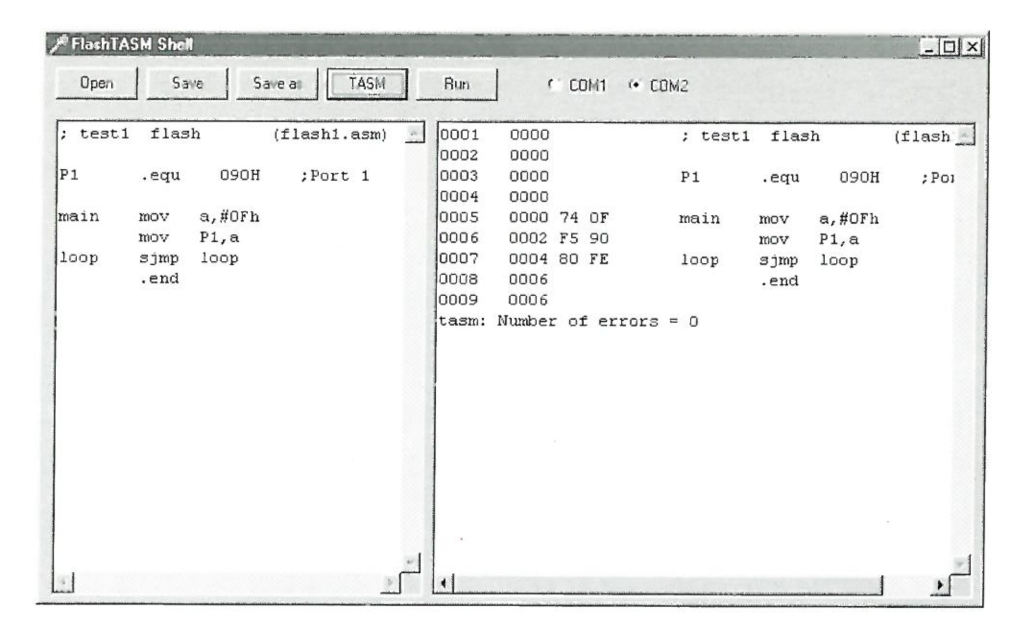



Figura 3. El primer programa de muestra en TASMedit.exe.

En nuestro primer experimento, todo lo que queremos hacer es conmutar los niveles de uno de los puertos de salida del microcontrolador. Después de todo, el primer paso de la automatización es una conmutación. También se puede observar el resultado utilizando un voltímetro conectado al pin P1.4 a través del conector K4 (ver Figura 2).

En lo que se refiere a cambiar el nivel en este pin, utilizaremos un pequeño programa en ensamblador. Brevemente, ensamblador es una notación utilizada para escribir instrucciones para un procesador o microcontrolador. Cada microcontrolador tiene un set de instrucción, el cual consta de valores numéricos y funciones asociadas. Las series siguientes de seis números representan un pequeño, pero completo, programa para un microcontrolador 89S8252. En realidad, es lo que llamamos un programa en código máquina:

```
116,15,245,144,128,254
```

En general, la costumbre es escribir programas de ordenador utilizando números hexadecimales en lugar de números decimales, porque así es más fácil de leer. En notación hexadecimal, el programa tiene un aspecto parecido a éste:

```
74 0F F5 90 80 FE
```

Nosotros tenemos que escribir esta secuencia de números en la memoria de programa del microcontrolador. Podemos usar un programa llamado MicroFlash para este propósito. Por último, el programa terminará al final de la memoria de programa del microcontrolador. El microcontrolador leerá los números de la memoria, uno después de otro, y después sabremos qué hacer. En lenguaje normal, lo podemos expresar de la siguiente forma:

**74:** Transferimos un valor numérico al acumulador (eso es mi memoria) — pero ¿cuál?
**0F:** Aquí es: OF — bueno, yo he hecho una nota de esto.
**F5:** OK, ahora tenemos que escribir el valor en un registro- ¿pero en cuál?
**90:** Ya veo, el registro para el puerto 1 está localizado en la dirección 90.
**80:** Y ahora tengo que hacer un salto corto, pero ¿adónde?
**FE:** Dos bytes atrás, desde la posición anterior.
**80:** El mismo salto de nuevo – OK, estaremos corriendo alrededor de un círculo.

Ésta es la forma en la que 'piensa' y actúa el microcontrolador, porque los ingenieros lo han aprendido así. Por último, un microcontrolador no es nada más que un circuito muy complejo hecho con puertas lógicas. Este circuito responde al estado de sus líneas de entrada, que en este caso son las líneas de datos conectadas a la memoria de programa de la unidad de procesamiento central.

Si queremos saber en detalle qué hay dentro de un microprocesador, tenemos que realizar un buen trabajo. Sin embargo, es suficiente para nosotros saber las instrucciones máquina del procesador y cómo utilizarlas. Esto significa que el microcontrolador es para nosotros como una caja negra que no conocemos y cuyo com-

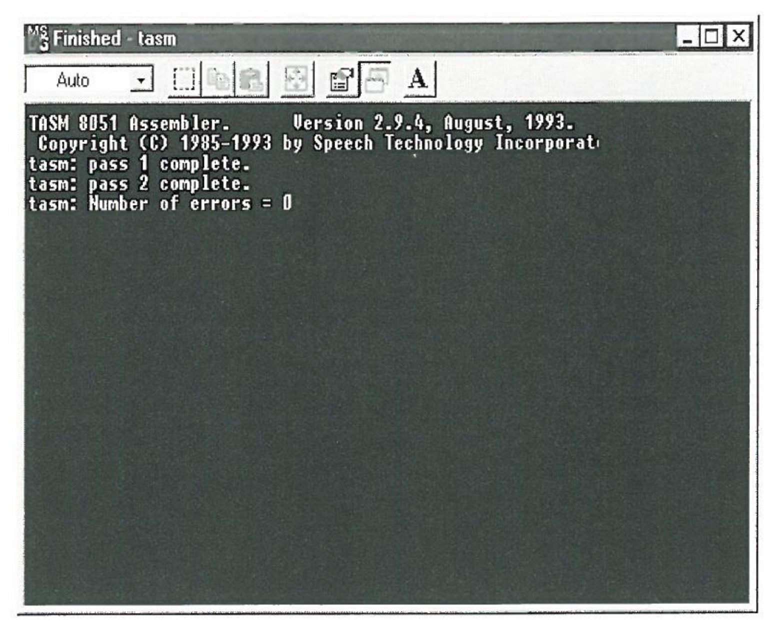



Figura 4. TASM en la ventana DOS.

portamiento es fácil de entender. Eso es como la moderna tecnología, es casi imposible alcanzar todos los niveles del conocimiento.

Ahora sabemos que un microcontrolador tiene su propio lenguaje, el cual realmente no consta de nada más que números. Sin embargo, obviamente hay un problema: el lenguaje que puede leer un microcontrolador se puede leer fácilmente y no es adecuado para personas. Nosotros no estamos acostumbrados a trabajar con números, sino con palabras. Consecuentemente, las palabras (las cuales se pueden recordar con más facilidad) se han empleado para representar de forma individual instrucciones del lenguaje máquina. El programador escribe esas palabras en un fichero de texto, y un programa especial las traduce en el lenguaje del microcontrolador. Este programa se llama ensamblador y el lenguaje de programación también se llama ensamblador (o lenguaje ensamblado). Ensamblador es una notación que podemos utilizar para decirle al microcontrolador qué debe hacer:

```
main    mov     a,#0Fh
        mov     090H,a
loop    sjmp    loop
```

Esto ya es mucho más legible. Realmente, nosotros sólo necesitamos saber dos palabras especiales: 'mov' y 'sjmp'. Ambas palabras se llaman nemónicas, es decir, marcadores utilizados en lugar de las instrucciones reales del lenguaje máquina. La palabra 'mov' (move) significa 'move', 'shift' o 'load'. A continuación ponemos dónde se va a cargar y después qué se va a cargar. En la primera línea, el valor numérico 0Fh (=15), se identifica con el símbolo '#', que significa carga en el acumulador A. El acumulador es un registro o memoria con un tamaño de ocho bits que puede tener valores numéricos entre 0 y 255.

En la segunda línea, el valor del acumulador se copia en la dirección 90h (=144). En esta dirección hay un registro cuyas líneas son llevadas al exterior del circuito integrado, en las líneas del Puerto 1. La palabra 'sjmp' (salto corto) causa un salto en la ejecución de un programa, en este caso la posición 'loop'. La palabra 'loop' se ha elegido de forma completamente arbitraria y simplemente representa una dirección, en este caso una posición en las series de instrucciones. El ensamblador trata tales palabras, las cuales se denominan etiquetas, como direcciones, y las sustituye con los valores numéricos adecuados. La instrucción sjmp puede dar lugar a un salto de 127 bytes hacia atrás o 128 bytes hacia delante. Un simple byte es suficiente para especificar el destino del salto. Aquí, el salto se calcula relativo a la posición normal del programa.

La palabra 'main' al comienzo del programa se elige también de forma arbitraria. Las únicas palabra clave de ensamblador son 'mov' y 'sjmp'. Nosotros podemos recordar fácilmente estas palabras sin la ayuda de una chuleta.

Sin embargo, aún hay un problema: ¿es realmente necesario recordar que un registro particular para el Puerto 1 está localizado en la posición 90h? Después de todo, sería mejor que pudiéramos escribir esto como texto. Esto se puede hacer fácilmente, simplemente tenemos que definir un pequeño trozo de texto como equivalente a un valor numérico. El ensamblador entonces sustituye este texto con el valor numérico correspondiente en cada posición en la que encuentra el texto. Para hacer tal definición utilizaremos la directiva de ensamblador 'equ'. Una directiva de ensamblador siempre comienza con una parada total, la cual informa al ensamblador que no es una instrucción en lenguaje ensamblador. En el siguiente ejemplo, la palabra 'P1' tiene asignado el valor 90h. En el programa real, el valor 90h se sustituye por 'P1':

```
; flash1.asm  port output

P1      .equ    090H ;Port 1

main    mov     a,#0Fh
        mov     P1,a
loop    sjmp    loop
        .end
```

Este listado también nos muestra algo más: al principio de la línea están localizadas las etiquetas de salto y nuevas palabras definidas, y todas las instrucciones del ensamblador están localizadas después. Por otro lado, también hay un comentario, el cual no forma parte de la traducción. Un comentario comienza con un punto y coma (;).

La notación especifica valores desde un ensamblador al siguiente. Aquí estamos utilizando el programa de libre distribución **TASM**. El TASM es muy sencillo y puede traducir programas para muchos tipos de microcontroladores diferentes, además tiene la tabla de instrucciones adecuada.

Al final del programa hay un bucle, en el que se produce un salto al bucle destino, siempre y por siempre. En otros lenguajes de programación, tal situación sería un bucle infinito y sin fin, lo cual es prácticamente equivalente a un fallo. En tal bucle, el procesador está supeditado y no puede salir de su control. En general, siempre debería estar claro una vez se han completado las cuestiones corrientes. Sin embargo, este bucle es muy importante. Sólo hay una cuestión, cambiando el estado del puerto. Si vamos a dejar al procesador con sus propios dispositivos, ejecutaremos comandos que estarán en la memoria de programa y quizás pertenezcan a un programa completamente dife-

rente. Consecuentemente, debemos de fijar un límite por medio de un bucle final. En realidad, el procesador está atrapado en este bucle, lo cual significa que sólo se puede salir de él mediante un RESET. Después de eso, el mismo programa debería comenzar de nuevo, o podríamos cargar un nuevo programa y después ejecutarlo. La carga de un programa tiene lugar en el estado de reset; la memoria de programa no se llena del todo, sino mediante funciones de bloque especiales en el microcontrolador que programa la flash ROM. Cada vez que la placa se enciende, se produce un reset corto de forma automática. Siguiendo esto, el microcontrolador encuentra el programa más recientemente cargado y lo ejecuta.

## Utilización del ensamblador TASM

Ha llegado la hora de hacer negocios. Lo que queremos hacer es escribir el primer programa, traducirlo y enviarlo a un microcontrolador. Para esto necesitamos algo de software. Nosotros utilizamos el ensamblador de libre distribución TASM, el cual podemos encontrar dentro del disco de trabajo en forma de fichero zip; también se puede descargar de la página de *Elektor Electronics*. El fichero TASM.ZIP se debe descomprimir en un directorio de trabajo situado en el disco duro que contiene el programa **TASMedit** y los programas de ejemplo.

Las características especiales del TASM permiten utilizar diferentes tipos de microcontroladores. Para cada tipo hay una tabla de instrucciones máquina disponible, las cuales se deben identificar cuando el programa se ejecute. Esto se hace utilizando un comando de línea; en este caso, nosotros utilizamos:

```
TASM -51 -b flash1.asm
flash1.bin
```

para especificar nuestro programa ejemplo particular, la tabla de instrucciones TASM51.tab y la salida en formato binario. Naturalmente, a nadie le gusta trabajar con las líneas de comando. Consecuentemente no usaremos todo esto en nuestro curso.

En general, queremos trabajar sólo con Windows, pero TASM es un pro-

# Software

Para cargar un programa en el microcontrolador de la Placa Flash 89S8252, necesitaremos el programa de Windows **MicroFlash.exe**, que puede encontrarse en la página de Elektor Electronics en las descargas de páginas gratuitas, (ver la lista para los artículos de Diciembre del 2001).
Debemos ejecutar el ensamblador TASM para el curso de programación, pero después necesitaremos también el Compilador de C Rigel READ51 y el Compilador de BASIC-52.
El ensamblador TASM es un programa popular, que se puede obtener junto con el TASMedit descargándolo de la página web de *Elektor Electronics*. Por favor, registre el software y pague al programador (T.N. Anderson) sus bien ganados honorarios. Para el Compilador de C no es necesario pagar nada, ya que Rigel está disponible de forma gratuita para uso educativo. Este compilador se puede obtener de la página web www.rigelcorp.com. BASIC-52 es un compilador de Basic interpretado creado por Intel, que está programado por máscara en una memoria de programa de un microcontrolador 80C52 y recibe la designación de 80C52-AH-BASIC. Este microcontrolador se ha utilizado durante casi dos décadas por ingenieros electrónicos y programadores y ha llegado a ser famoso internacionalmente, debido principalmente a los artículos de Elektor Electronics. Hace varios años, Intel cesó la producción de este circuito integrado, pero dejaron el lenguaje de programación abierto, convirtiéndose en un lenguaje de uso general. El lenguaje se ha desarrollado y adaptado para poder ser utilizado por otros microcontroladores. Probablemente la versión más actualizada es la V1.3, que fue presentada en Febrero de 2001 en nuestra revista y está disponible en el Servicio de Lectores o en el disco (código de pedido **000121-11**).
El disco para este curso (disponible en el Servicio de Lectores con el código **010208-11**) contiene el ensamblador TASM, el TASMedit y los primeros programas de ejemplo, junto con BASIC-52, MicroFlash.exe y un pequeño programa emulador de Basic con sus propios programas de ejemplo.

# Formatos numéricos

La realidad es que utilizamos un sistema de numeración decimal debido a que tenemos diez dedos. El sistema de números naturales para un sistema por ordenador es el binario. El sistema hexadecimal representa un compromiso, en el cual el rango de números va desde 0 a 15, con la incomprensible representación de caracteres para los números que pasan por encima de 9 y que son las letras A, B, C, D, E y F

| Decimal | Hexadecimal | Binario |
|---|---|---|
| 0 | 00h | 00000000b |
| 1 | 01h | 00000001b |
| 2 | 02h | 00000010b |
| 3 | 03h | 00000011b |
| ... | ... | ... |
| 10 | 0Ah | 00001010b |
| 11 | 0Bh | 00001011b |
| 12 | 0Ch | 00001100b |
| 13 | 0Dh | 00001101b |
| 14 | 0Eh | 00001110b |
| 15 | 0Fh | 00001111b |
| 16 | 10h | 00010000b |
| 17 | 11h | 00010001b |
| ... | ... | ... |
| 253 | FDh | 11111101b |
| 254 | FEh | 11111110b |
| 255 | FFh | 11111111b |

En los programas en ensamblador, generalmente es posible elegir el tipo de notación que queremos utilizar. Si estamos describiendo el estado de un puerto de 8 bits, la notación binaria es particularmente clara. Por ejemplo, el bit de más a la derecha es el P1.0 y el de más a la izquierda es el P1.7. aquí, ocho pines requieren ocho bits, o un byte.

Cuando TASM traduce un programa, podemos especificar el formato en el cual se almacenan los resultados. En formato binario, sólo se escriben las instrucciones individuales de lenguaje máquina. Un editor de texto no puede hacer nada en tal fichero.
El formato Intel hex utiliza líneas de texto que contienen números en hexadecimal. Además del código real, hay una dirección de inicio y una comprobación en cada línea. Este formato también se puede ver como texto.

grama puro en DOS. Por esta razón, se ha escrito una interface Windows para el programa, llamada TASMedit.exe. Ésta incluye su propio editor y permite al usuario ver inmediatamente el resultado de la traducción, incluyendo los posibles mensajes de error. La herramienta flash de descarga para la 'Placa Flash de Elektor Electronics' también está integrada en este programa. Hemos intentado hacer las cosas tan fáciles como sea posible para todos los participantes del curso.

El programa tiene dos ventanas de texto (ver **Figura 3**). En la de la parte izquierda está el editor de texto. Se puede utilizar para introducir de forma manual un programa o cargar un programa desde el disco duro. El botón TASM arranca el ensamblador en segundo plano. La ventana de la derecha representa la lista de ficheros en ensamblador junto con algunos mensajes de error que pueden aparecer. En esta ventana también se puede ver el proceso de descarga de un programa cuando se está enviando a la placa del microcontrolador.

Pulsando sobre el botón TASM, primero generamos un fichero llamado Work.asm, el cual posee el contenido normal de la ventana del Editor. Después este texto se traduce. TASM se llama desde la interface Windows utilizando la línea de comandos:

```
TASM -51 -b -work.asm work.bin
```

Esto significa que el formato binario siempre se utiliza aquí. El resultado de la traducción se almacena en un fichero llamado Work.bin que es el fichero que se lee por el módulo de descarga cuando se actúa sobre el botón de RUN. El ensamblador también genera un fichero llamado Work.list, que contiene el fichero listado, el cual mantiene la traducción en un formato legible. Al realizar experimentos con el ensamblador tenemos la ventaja de que siempre se utilizan los mismos nombres de ficheros para la traducción, porque el texto fuente sólo se tiene que salvar una vez después de haber completado toda la operación con éxito. Esto significa que no tenemos en el disco duro toda la colección de datos basura de todas las operaciones que no sirven, sino que sólo se salva el texto fuente y los ficheros de trabajo de las operaciones más recientes. Si olvidamos salvar la última versión del fichero de texto, o si el PC se cuelga mientras estamos trabajando, siempre podemos utilizar el fichero Work.asm para recuperar los frutos de nuestro trabajo.

Cuando se ejecuta TASM de forma automática desde la interface Windows, aparece en una ventana de DOS (Ver **Figura 4**). Esta ventana se debe cerrar antes de que podamos seguir. Al principio, puede ser muy entretenido ver como trabaja TASM, pero después de la tercera vez resultará un poco molesto para los usuarios acostumbrados al Windows. Por ello, desde ahora diremos que tendremos una ventana que se cierra de forma automática. Esto no es problema, porque Windows proporciona una solución. Primero ejecutaremos el fichero TASM.EXE con el botón derecho del ratón y abriremos el menú de Propiedades (Properties). Dentro de Properties/Program encontraremos la opción 'Close when done' (Cerrar cuando ejecute). Habilitando esta opción (ver **Figura 5**), la ventana genera un enlace en forma de un fichero llamado TASM.PIF. Desde ahora, la ventana de DOS se cerrará automáticamente después de que TASM haya realizado su trabajo.

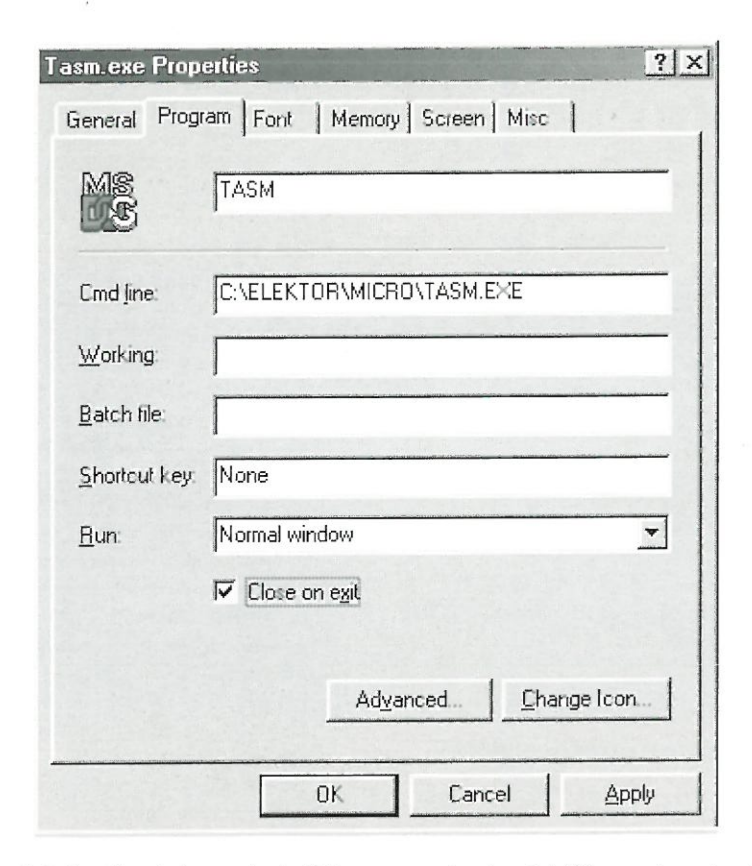



Figura 5. Selección de la propiedad 'Cerrar cuando ejecute' (Close when done).

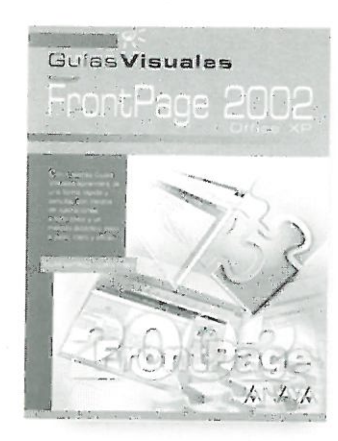
Una vez que el programa ha traducido con éxito, se puede utilizar el botón RUN para transferirlo al sistema 'Flash Board' (Placa Flash) y ejecutarlo. Para esto, tenemos que seleccionar un puerto PC COM y conectarlo al conector de programación de la placa (K2). Si reúne todos los requisitos más importantes (aplicando la tensión de alimentación a la Flash Board) podrá ejecutar el programa de prueba. En el caso de nuestro primer ejemplo, todo lo que tenemos que hacer es observar el estado de las salidas del puerto 1 para ver si el resultado ha sido satisfactorio. Utilizando un medidor de alta impedancia, deberíamos ver casi siempre 5 V exactos en los pines del puerto P1.0 a P1.3 y casi 0 V en los pines P1.4 a P1.7 (para ser más preciso, unos 30 mV).

En el terminal de masa, sin que se esté ejecutando el programa, o después de un proceso de reset, todos los pines del puerto tienen un estado alto, con una tensión de 5 V en cada pin. Esto se puede comprobar fácilmente utilizando un osciloscopio o un polímetro. El nuevo programa cargado y ejecutado cambia el estado de cuatro líneas: P1.4 a P1.7 deberían estar ahora a nivel bajo, lo cual significa que su tensión está alrededor de los 9 V, mientras que P1.0 a P1.3 debe permanecer a nivel alto. El programa ha transferido el valor 15 (=0Fh) al puerto 1. El estado de este bit se puede ver en los pines del puerto.

(010206-2)

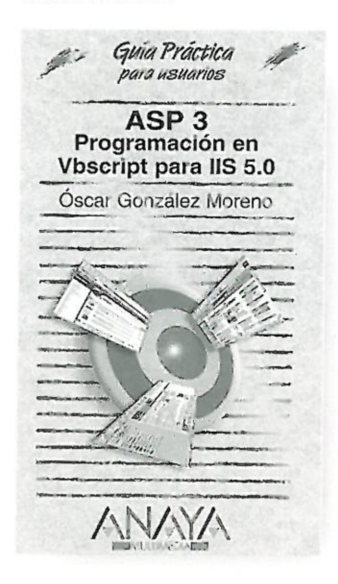
*Concluimos así nuestra breve introducción al trabajo en ensamblador. En la siguiente entrega del curso discutiremos pequeños programas de muestra que nos sirven para investigar las propiedades del puerto del microcontrolador. Veremos las entradas, salidas y velocidades que alcanza.*

**FrontPage 2002**
**Por Julián Martínez y Elvira Yebes.**
**ISBN 84-415-1196-9**
**192 páginas**
**Editorial Anaya Multimedia**




La herramienta de creación y administración de sistios Web, Microsoft FrontPage 2002, ofrece a los usuarios nuevas y potentes características, ayudando a individuos, equipos y organizaciones a controlar y administrar sitios Web con sencillez y seguridad, y ofreciendo funciones de integración con otras aplicaciones Office XP.
Este libro está dirigido a todas aquellas personas que necesitan iniciar sus conocimientos sobre FrontPage 2002. Su estructura está pensada para que pueda ser utilizado bien como guía de aprendizaje de las técnicas básicas del funcionamiento del programa, o bien como guía de referencia rápida para llevar a cabo cualquier tarea que el lector necesite realizar.
La Guía Visual de FrontPage 2002 se encuentra estructurada en capítulos que engloban operaciones relacionadas entre sí o con características comunes.
En ellos aprenderá operaciones tales como: crear un sitio Web utilizando los asistentes y plantillas, insertar texto, imágenes, tablas, listas, sonidos..., administrar un sitio Web, crear hipervínculos a otros documentos, crear páginas Web con marcos, insertar formularios, publicar el sitio Web creado con FrontPage 2002.

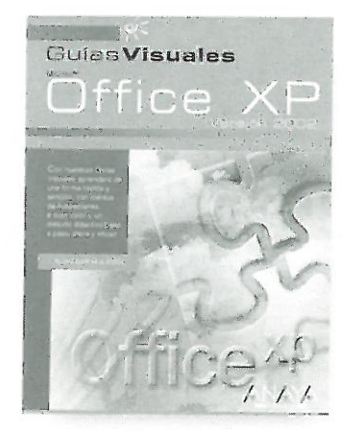
**ASP3. Programación en Vbscript para IIS 5.0**
**Por Óscar González.**
**ISBN 84-415-1157-8**
**352 páginas**
**Editorial Anaya Multimedia**




La tecnología de páginas activas ASP ha sido y es una de las más populares en el terreno de aplicaciones para entornos I*Net (Internet, Intranet y Extranet). Miles de empresas tradicionales y .COM las utilizan para sus desarrollos en todo el mundo, y de hecho es una de las tecnologías con más auge y aceptación en este mercado.
Esta guía ofrece una visión eminentemente práctica sobre el desarrollo de sitios Web con ASP 3. Al mismo tiempo presenta, de forma general, los sistemas en los que se basa ASP: servidor Internet Information Server 5.0, Windows Scripting Host, objetos COM, y acceso a motores de bases de datos Access y MS SQL Server 2000.
El libro hace una pequeña introducción al lenguaje XHTML, sucesor directo de HTML, y que utilizaremos en nuestros ejemplos como lenguaje de creación de páginas Web. Destacan también los apuntes sobre tecnologías de nuevo acuño, como el uso del lenguaje de marcado XML dentro de páginas ASP, o la tecnología integrada en la plataforma Microsoft .NET.
Al término de la guía, el lector habrá hecho un coherente recorrido que comienza en la instalación, puesta en marcha, sintaxis básica y entorno de desarrollo del programa y finaliza con las características más avanzadas del programa. Esto le permitirá manejar el tema con solvencia y afrontar por sí mismo la implantación de un sitio Web dinámico y original en el competitivo entorno de hoy.
Contenido del libro:

- Conceptos básicos.
- Instalación y trabajo con Internet.
- Information Server 5.0.
- Sintaxis de ASP (VBScript, JScript).
- Introducción a XHTML.
- ASP avanzado.
- Sistemas de objetos COM+.
- Acceso a bases de datos (ODBC y ADO).
- ASP y XML.

**Office XP**
**Por Pedro García Alegría.**
**ISBN 84-415-1187-X**
**192 páginas**
**Editorial Anaya Multimedia**




La nueva versión de Office, XP, es la mejor suite ofimática para que profesionales, pequeños negocios, grandes y medianas empresas, y usuarios en general, puedan extraer la información de su entorno y convertirla en valor real para su trabajo.
Este libro está dirigido a todas aquellas personas que necesitan iniciar sus conocimientos sobre Office XP. Su estructura está pensada para que pueda ser utilizado tanto como guía de aprendizaje de las técnicas básicas del funcionamiento del programa, así como guía de referencia rápida para llevar a cabo cualquier tarea que el lector necesiterealizar con las distintas herramientas.
La Guía Visual de Office XP se encuentra estructurada en capítulos que explican las diferentes herramientas que componen este paquete, así como sus características comunes, como por ejemplo la instalación y mantenimiento, manejar el Portapapeles de Office, familiarizarse con el entorno, o utilizar la ayuda.
Entre los principales contenidos que se recogen destacan la redacción y edición de textos en Word, el diseño de preentaciones en PowerPoint, creación de libros en Excel, y el trabajo con bases de datos en Access. También se explica el uso de aplicaciones compartidas: Microsoft Document Imaging, para la digitalización y reconocimiento óptico de caracteres; creación de páginas Web con FrontPage o con Word; gestionar el correo electrónico con Outlook; o intercambio de información y trabajo en colaboración entre las distintas herramientas.

Contenido del libro:

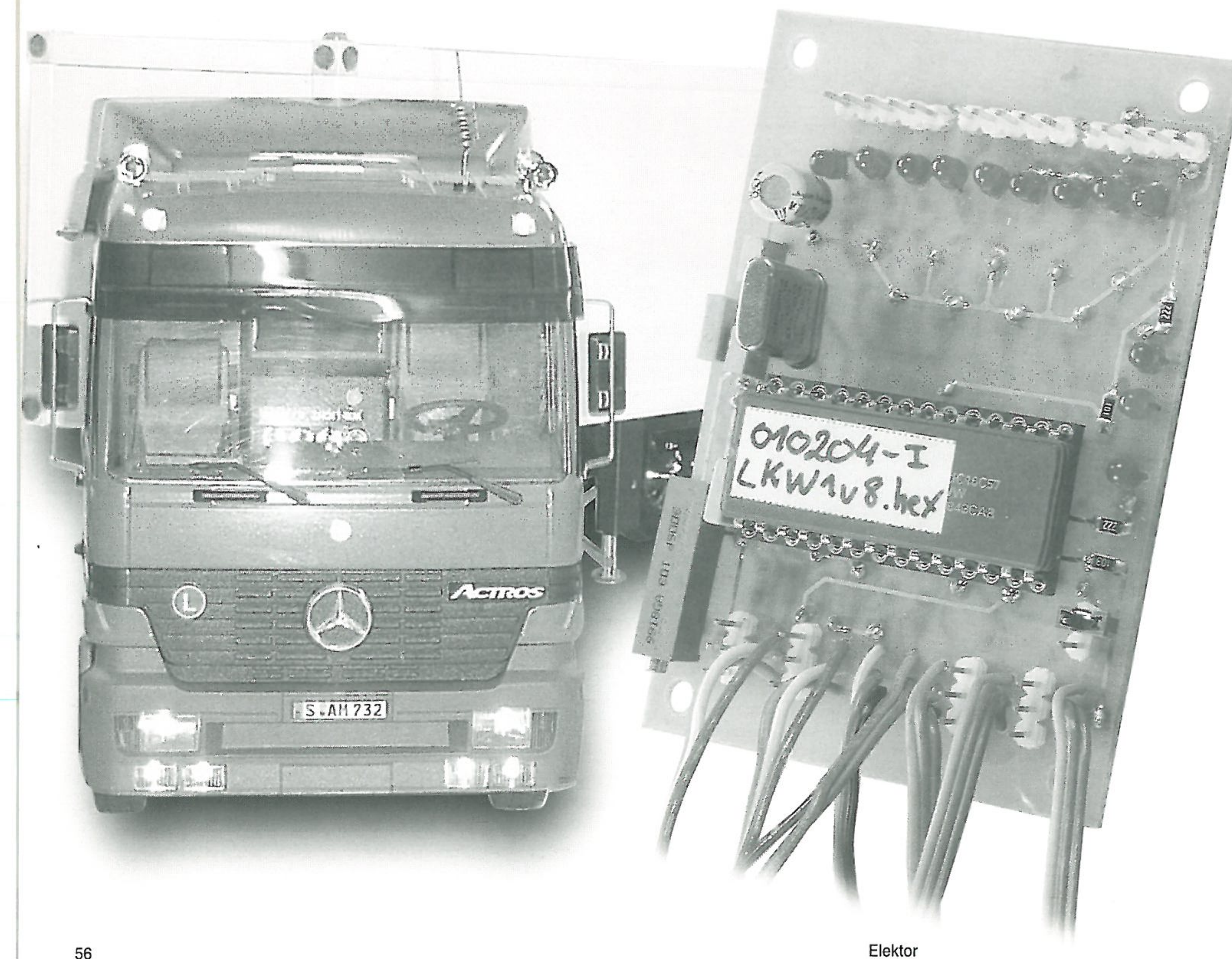
- Aspectos generales.
- Microsoft Word.
- Microsoft Excel.
- Microsoft Access.
- Microsoft PowerPoint.
- Otras funciones de Office XP.
- Transferencia de información e Internet.
- Microsoft FrontPage y diseño web.
- Microsoft Outlook.
- Glosario.

# Iluminación y caja de cambios

## para camiones de modelismo por control remoto

Diseñado por D. Dzida

**El circuito descrito aquí utiliza un microcontrolador PIC para tomar el control completo de la iluminación y la caja de cambios de un camión de modelismo. También puede utilizarse con grúas, excavadoras o vehículos similares.**



## Datos técnicos

| | |
|---|---|
| Tensión de alimentación: | 5V (desde receptor) |
| Consumo de corriente: | menor de 10 mA (excluyendo corriente de bombillas) |
| Tensión de alimentación externa: | 5–12V |
| Velocidad de parpadeo: | fija, aprox. 0.5 Hz |
| Desplazamiento servo caja cambios: | ajustable usando un potenciómetro |
| Acoplamiento del trailer servo dir. De rotación: | seleccionable (izquierda/derecha) usando JP1 |
| Corriente salida máx.: | 0.3 A |

Salida para control sólo (max. 10 mA)

Relés se pueden conectar (sólo con diodo en inverso!)

**Nota para alcanzar la máxima corriente de bombilla:**
Si las bombillas se alimentan desde una fuente interna (ver Figura 5) utilizaremos el controlador de velocidad del receptor BEC, no debiéndose exceder la corriente de salida máxima del controlador de velocidad BEC.

El problema con esta clase de vehículos controlados remotamente es que tienen un gran número de funciones, lo cual significa que debe utilizarse una unidad de control remoto cara. El proyecto que presentamos aquí consta de una pequeña placa de circuito impreso, con un microcontrolador de la popular familia PIC de Microchip, el cual tiene funciones extendidas. Esto supone una combinación de la función automática de control y las funciones de decodificación suplementarias.

Las funciones automáticas son :

- Intermitentes que se activan de forma automática cuando el vehículo gira.
- Luces de freno que se activan cuando el acelerador se deja de accionar.

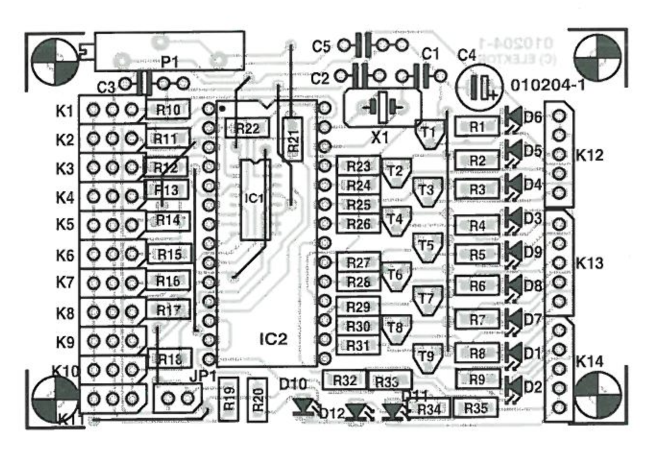
Figura 1. Esquema del circuito de control de luces y caja de cambios. El microcontrolador evalúa las señales de pulso del receptor de control remoto y genera señales adicionales para las luces y caja de cambios.

## BEC

El acrónimo 'BEC' significa 'Circuito Eliminador de Batería'. Éste es un circuito que elimina la necesidad de una batería separada para alimentar el control remoto proporcionando una tensión de alimentación procesada adecuadamente (estabilizada y desacoplada) desde el pack de baterías del motor. En un camión de modelismo, este BEC está en el controlador de velocidad (el controlador de velocidad del motor en la cabeza tractora). Tal controlador de velocidad también incluye un BEC con una tensión de salida de 5 V. En el caso de los reguladores especiales para camiones, esta salida de 5 V puede alimentar una carga más pesada que la de los reguladores normales de coches. La corriente que puede alimentar el regulador depende del modelo, pero rondará entre 1 y 3 Amperios.

– Luces posteriores que se encienden cuando el vehículo va marcha atrás.

Además, la unidad de control remoto puede utilizarse para conmutar las luces principales, luces de emergencia, la luz de la cabina, el claxon y canales de salida adicionales.



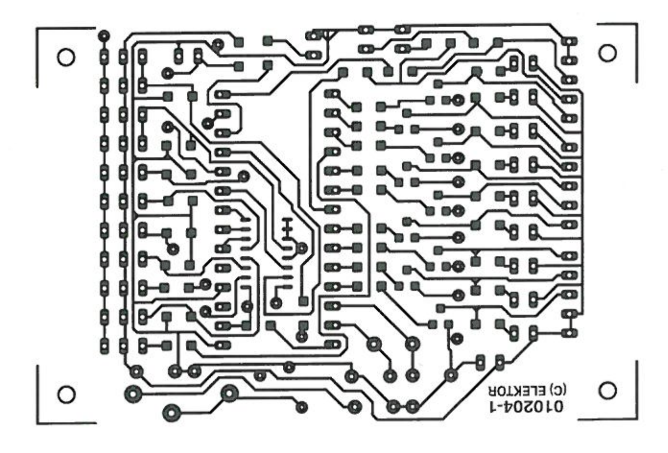




Figura 2. Placa de circuito impreso por las caras de pistas y componentes. La placa es a simple cara.

## Corriente de lámpara

El consumo de corriente del circuito sin ninguna lámpara conectada es de al menos 10 mA. Cada una de las nueve salidas conectada a los transistores puede conmutar a un máximo de 300 mA, lo que hace que para todas tengamos 9 x 0,3 = 2,7 A. Esta cantidad de corriente sólo se puede proporcionar por una alimentación externa (cableada, tal y como se muestra en la Figura 6).
La tensión de alimentación externa de 5 V está cableada como se muestra en la Figura 5. La salida de 5 V del controlador de velocidad del vehículo no sólo debe proporcionar la corriente para el receptor de control remoto y los servos, sino también la corriente total de las bombillas. Por ello, si hay muchas lámparas, deberíamos utilizar un controlador de velocidad que proporcione 1,5 A o más. En la siguiente lista vemos las distintas lámparas seleccionadas (lámparas de 5 V/40 mA):

| | | | |
|---|---|---|---|
| Luces cruce: | 6 × 40 mA | = | 240 mA |
| Luz principal: | 2 × 80 mA | = | 160 mA |
| Luz frenado: | 4 × 40 mA | = | 160 mA |
| Luces posteriores: | 2 × 40 mA | = | 80 mA |
| Intermitentes: | 6 × 40 mA | = | 240 mA |
| Luces emergencia: | 2 × 40 mA | = | 80 mA |
| Cambio canal: | 1 × 80 mA | = | 80 mA |
| Corriente total bombilla: | | | 1.04 A |

Por otro lado, el servo de la caja de cambios puede desplazarse arriba o abajo mediante un conmutador o el mando de control (como en un Porche real, con sólo tres marchas...).

Por último, hay una función trailer,

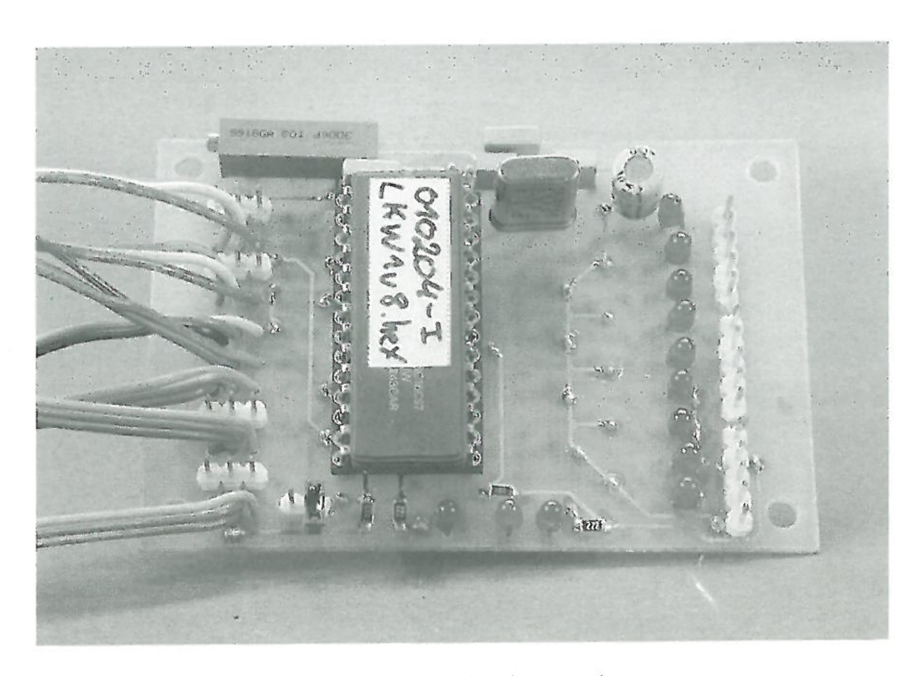



Figura 3. Vista superior de la placa de circuito de prototipos.

**LISTA DE MATERIALES**

**Resistencias:**
R1-R9,R20,R34,R35 = 2k2
R10,R12,R21,R32,R33 = 100Ω
R11,R13-R19,R22 = 10kΩ
R23-R31 = 1k2
P1 = 10 KΩ potenciómetro de montaje horizontal

**Condensadores:**
C1,C2 = 22pF
C3,C5 = 100nF
C4 = 10µF 16V radial

**Semiconductores:**
D1-D9,D11,D12 = LED rojo, 3 mm, alta eficiencia (baja corriente)
D10 = LED verde, 3 mm, alta eficiencia (baja corriente)
T1-T9 = BC817
IC1 = 4051 SMD
IC2 = PIC16C57, programado, código de pedido **010204-41**

**Varios:**
JP1 = 2-terminales molex macho con puente
K2,K4-K7,K9,K11 = cable de conexión a servo
K1,K3,K8,K10 = conector de 3 pines
K12,K13,K14 = conector de 5 pines
X1 = cristal de cuarzo de 16 MHz
PCB, código de pedido **010204-1**
Disco, ficheros con código fuente, código de pedido **010204-11**

en la cual un simple servo opera el acoplamiento del tractor y la longitud del soporte asociado al trailer. Esta doble función está especialmente diseñada para su uso con el kit del camión Tamiya. Si sólo es necesario el acoplamiento del trailer, esta función se puede utilizar también para otros modelos de camiones.

## Funcionamiento del circuito

El corazón del circuito del controlador es un PIC16C57 de Microchip a 16 MHz. La versión viene en encapsulado de DIL28, ya que esto hace más fácil llevar fuera el software o sus ampliaciones. El circuito, que se muestra en el esquema de la Figura 1, asume el uso de un circuito eliminador de batería (BEC) mediante una tensión de alimentación regulada de 5 V.

El microcontrolador evalúa las señales de siete canales de salida del receptor de control remoto. En cada una de esas salidas, el receptor proporciona la conocida señal de salida modulada en anchura de pulsos con rangos que van de 1 a 2 ms (1,5 ± 0,5 ms). Esas señales, que controlan la caja de cambios, la intensidad de luz, la conmutación de canal, luces de emergencia, ruedas y motor, están conectadas a K2, K4, K5, K6, K7, K9 y K11, respectivamente. Desde aquí, alcanzan las entradas X0-X6 del multiplexor (IC1), y desde el multiplexor llegan a los pines RA0-RA3 del microcontrolador PIC (IC2). El multiplexor CMOS 4051 se puede omitir en principio, pero entonces sería necesario un PIC más potente y grande.

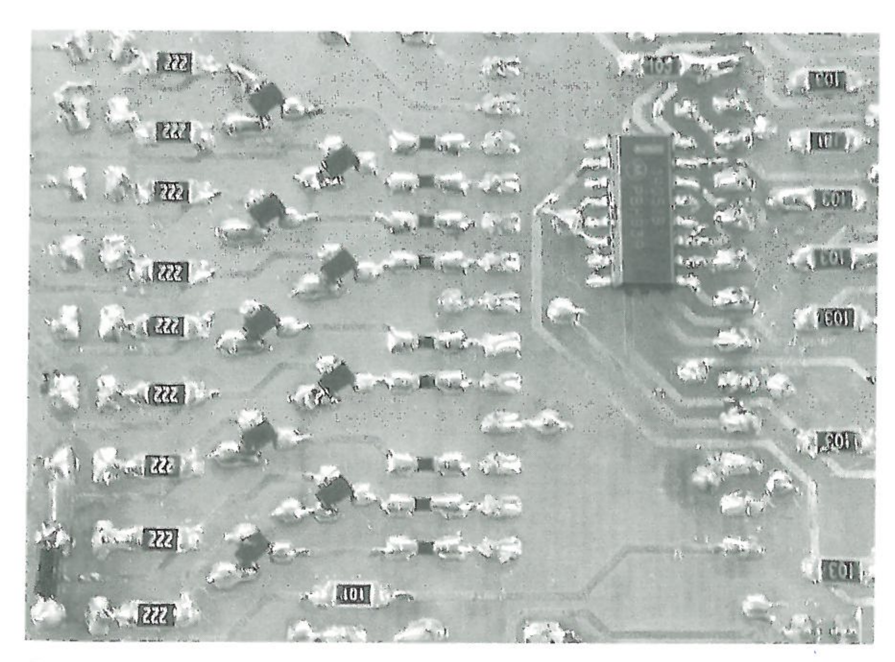


Figura 4. Los componentes SMD se fijan en la parte inferior de la placa de circuito impreso.

Las señales para las funciones controladas directamente por el controlador remoto (motor y ruedas) no sólo pasan al microcontrolador, sino que también se conectan directamente a los conectores de salida para la conexión al servo de las ruedas (K8) y el controlador de velocidad del motor (K10).

El pin del puerto RB0 del microcontrolador está conectado al puente (jumper) JP1. Este puente selecciona la dirección de rotación del servo para el acoplamiento del trailer. El recorrido del servo de los cambios se puede fijar utilizando P1, el cual se conecta al pin del puerto RB1.

En el proceso de programación de varias señales de entrada, el microcontrolador genera señales de control para dos salidas de servo (K1: servo de caja de cambios y K3: servo del acoplamiento del trailer) y once salidas conmutadas que están disponibles en los conectores K12, K13 y K14 Se implementan nueve salidas conmutadas como salidas en colector abierto mediante transistores externos (T1-T9), mientras que dos pines de salida (de K14) están conectados directamente a los pines del puerto RC3 y RC4. La **Tabla 1** nuestra un resumen de los conectores con sus designaciones y funciones. Si está interesado en el software que procesa las señales en el microcontrolador PIC, puede descargarlo gratuitamente de la página web de Elektor Electronics.

Cada una de las salidas conmutadas dispone de un LED que indica el estado de la salida (D1-D9 y D11). El LED D10 tiene una función especial. Después de que la tensión de alimentación está conectada, se calibran algunas de las señales del microcontrolador (como las señales para los intermitentes, las luces de frenado y las de marcha atrás). El mando de control no se debe mover mientras esto sucede. El LED D10 se ilumina sólo después de que la inicialización se haya completado con éxito (lo cual tiene lugar después de cuatro segundos) para indicar que el circuito está listo para su uso. El LED D12 está conectado a un canal conmutado que no se utiliza normalmente (K14/3) y no tiene ninguna función activada. Sin embargo, el autor intenta usar

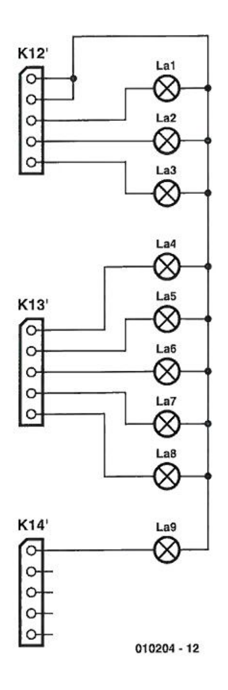



Figura 5. Conexiones de la lámpara cuando utilizamos una tensión de alimentación interna.

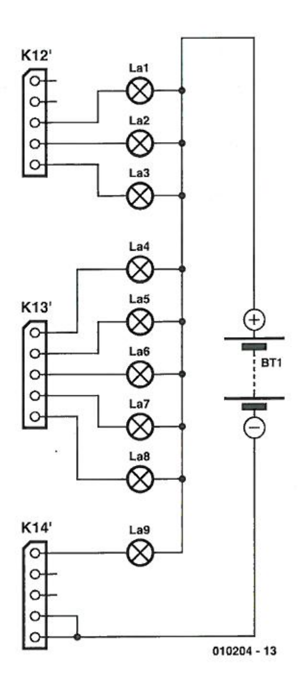



Figura 6. Conexiones de la lámpara cuando utilizamos una tensión de alimentación externa (5–12 V).

esta salida para soportar un enlace de infrarrojos entre la cabeza tractora y el trailer.

## Construcción y uso

Una vez fijados todos los componentes sobre la placa de circuito impreso (Figura 2) se requiere alguna atención especial. Aunque la

# Tabla I. Sumario de conexiones y funciones

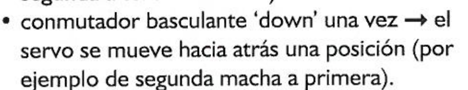
### Caja de cambios: K1 y K2

K1: Conexión servo caja de cambios (sólo para una caja de 3 velocidades)

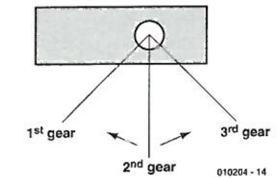
K2: Conexión para el cable del receptor 'salida' caja de cambios.
Función de la unidad de control remoto: conmutador basculante con tres posiciones: up / off / down

**Función:**

El recorrido del servo a la izquierda o derecha de la posición central se puede establecer con P1. Cualquier cambio en P1 llega a ser efectivo sólo después de que la alimentación se activa. Cuando se conecta la alimentación, el servo siempre se mueve a la posición central (segunda marcha).



- conmutador basculante 'up' una vez → el servo se mueve hacia una posición (por ejemplo de segunda a tercera marcha) .
- conmutador basculante 'down' una vez → el servo se mueve hacia atrás una posición (por ejemplo de segunda macha a primera).



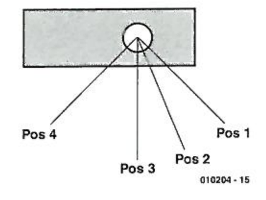

### Acoplamiento del trailer: K3 y K4

K3: Conexión para el servo de acoplamiento (para un camión Tamiya que funciona con soporte eléctrico: ver texto)

K4: Conexión para el cable de la salida del receptor 'acoplamiento del trailer'.
Función de la unidad de control remoto: conmutador basculante con tres posiciones: up / off / down

**Función:**

- conmutador basculante 'up' → el servo se mueve a la posición 1 (en esta posición, el soporte se baja hasta un límite en el que el trailer se suelta).
- conmutador basculante 'off' → el servo se mueve a la posición 3 (si las luces de marcha atrás no están, no se puede abrir el enganche).
- conmutador basculante 'down' y luces de atrás en ON → el servo se mueve a la posición 2 (en esta posición, el soporte se baja hasta un límite en el que el trailer se suelta).
  Después, si no se actúa sobre el conmutador basculante (retorna automáticamente a la posición central), el servo se mueve a la posición 4. El acoplamiento del trailer ahora se abre y la cabeza tractora del camión se separa del trailer. El acoplamiento permanece abierto hasta que se coloque en la posición 'up' (donde alcanza el soporte y cierra el acoplamiento).




### Luz de cruce/ luz principal: K5

K5: Conexión para el cable de salida del receptor luz de cruce/ luz principal
Función de la unidad de control remoto: cambiar a conmutador de palanca de tres posiciones: up / off / down

**Función:**

- conmutador de palanca en posición 'up' → todas las luces activas (K12/3&4)
- conmutador de palanca en posición 'off'→ luces de cruce activas (K12/3)
- conmutador de palanca en posición 'down' → luz de cruce y principal activas (K12/3&4)

### Conmutación/manipulación de canal: K6

K6: Conexión para el cable de salida del receptor **conmutación / manipulación de canal.**
Función de la unidad de control remoto: conmutador basculante con tres posiciones: up / off / down

**Función:**
- conmutador basculante 'up' una vez → canal manipulado on (K14/1)
- conmutador basculante de nuevo 'up' → canal manipulado off (K14/1)
- conmutador basculante 'down' → canal conmutado on (K14/2)
- conmutador basculante 'off' → canal conmutado off (K14/2)

**Nota importante:**
La conmutación de canal sólo es adecuada para propósitos de control (max. 10 mA), no conecte ninguna lámpara o relé sin utilizar etapas de driver adecuadas.

### Luces de emergencia: K7

K7: Conexión al cable de salida del receptor 'señal / luces emergencia'.
Función de la unidad de control remoto: conmutador basculante con tres posiciones: up / off / down

**Función:**
- conmutador basculante 'up' una vez → luces de emergencia on (parpadeo R/L, K13/4&5)
- conmutador basculante de nuevo 'up' → luces de emergencia off
- conmutador basculante 'down' una vez → señales intermitentes on (K13/2/3)
- conmutador basculante de nuevo 'down' → señales intermitentes off

### Indicadores L/R y servo de ruedas: K8/K9

K8: conexión a servo de ruedas
K9: conexión para el cable de la salida ruedas del receptor
Función de la unidad de control remoto: control de ruedas con el mando

**Función:**
Después de aplicar la tensión de alimentación, la posición neutra del mando de control se almacena. No moveremos el mando de control hasta que el LED D10 se ilumine.

- mando de control a la izquierda → indicador de izquierda en on (K13/5)
- mando de control en el centro → todos los indicadores off
- mando de control a la derecha → indicador de giro a la derecha on (K13/4)

### Luces de freno y posteriores y controlador de velocidad: K10/K11

K10: conexión para el controlador de velocidad
K11: conexión para el cable de salida 'controlador de velocidad' del receptor
Función de la unidad de control remoto: mando de control adelante / atrás

**Función:**
Después de que se haya aplicado la tensión de alimentación, se almacena la posición central del mando. No moveremos el mando de control hasta que el LED D10 se ilumine.

- mando de control en posición central → luces de freno on (K12/5)
- mando de control en posición adelante → luces de freno y posteriores off
- mando de control en posición atrás → luces posteriores on (K13/1)
- mando de control se mueve de atrás a la posición central → luces de freno y posteriores on

## Direcciones web

| | |
|---|---|
| *www.robbe.de* | todo tipo de productos |
| *www.multiplex-rc.de* | todo tipo de productos |
| *www.schulze-elektronik.com/index_uk.htm* | reguladores, cargadores |
| *www.kontronik.com/gate_engl.htm* | reguladores, motores |
| *www.hacker-motor.com/english/englisch.html* | motores |
| *www.wedico.de/index_eng.html* | camiones de modelismo |

placa del circuito está a simple cara, los componentes están montados en ambas caras. IC1 (un 4051 en encapsulado SMD) está soldado en la cara de pistas. Para fijar los componentes en la otra cara, es mejor comenzar con los puentes. Debemos prestar atención a la colocación que tienen los componentes en la placa de prototipos en las fotos (**Figuras 3 y 4**). Ésta es la placa del circuito final que sufrió alguna pequeña modificación del prototipo original, pero eléctricamente, las dos placas de circuito son iguales.

Toda la información esencial en lo que a disposición de pines en el conector, cableado y uso de la placa de circuito impreso se refiere, puede verse en la **Tabla 1**, con una excepción: la conexión de las bombillas que se muestra en las **Figuras 5 y 6**. Las salidas RB5-RB7 y RC0-RC7 del PIC controlan directamente las luces a través de los transistores BC817. Debido a que esas salidas están implementadas como salidas en colector abierto, las bombillas se pueden conectar directamente a la placa del circuito (como en la Figura 5) o a través de una tensión de alimentación externa (utilizando una batería separada, como se muestra en la **Figura 6**). En el último caso, un terminal de cada bombilla se conecta al terminal positivo de la tensión de alimentación externa, la cual puede tener un rango de 5-12 V. Cuando conectamos las bombillas, debemos tener especial cuidado en no hacer ningún cortocircuito entre la tensión de alimentación de +5 V en los pines 1 y 2 de K12 y la tensión de alimentación externa. Tal cortocircuito destruiría completamente al PIC y el receptor de control remoto.

Obsérvese que si las bombillas están conectadas directamente a la placa de circuito impreso (como se muestra en la Figura 5), toda la corriente de las lámparas se debe proporcionar a través del controlador de velocidad. Los reguladores especiales de camiones pueden proporcionar más corriente en la salida BEC (5 V) que los reguladores de coche normales. En cualquier caso, la disposición de las bombillas se debe elegir de forma que la alimentación BEC no sufra sobrecarga (ver la 'Corriente de bombilla' en el lateral). Los relés también se pueden conectar a las salidas en colector abierto, pero sólo si se fijan por medio de diodos en inverso. La corriente máxima por la bobina del relé está sujeta a las mismas consideraciones que la corriente máxima de la bombilla.

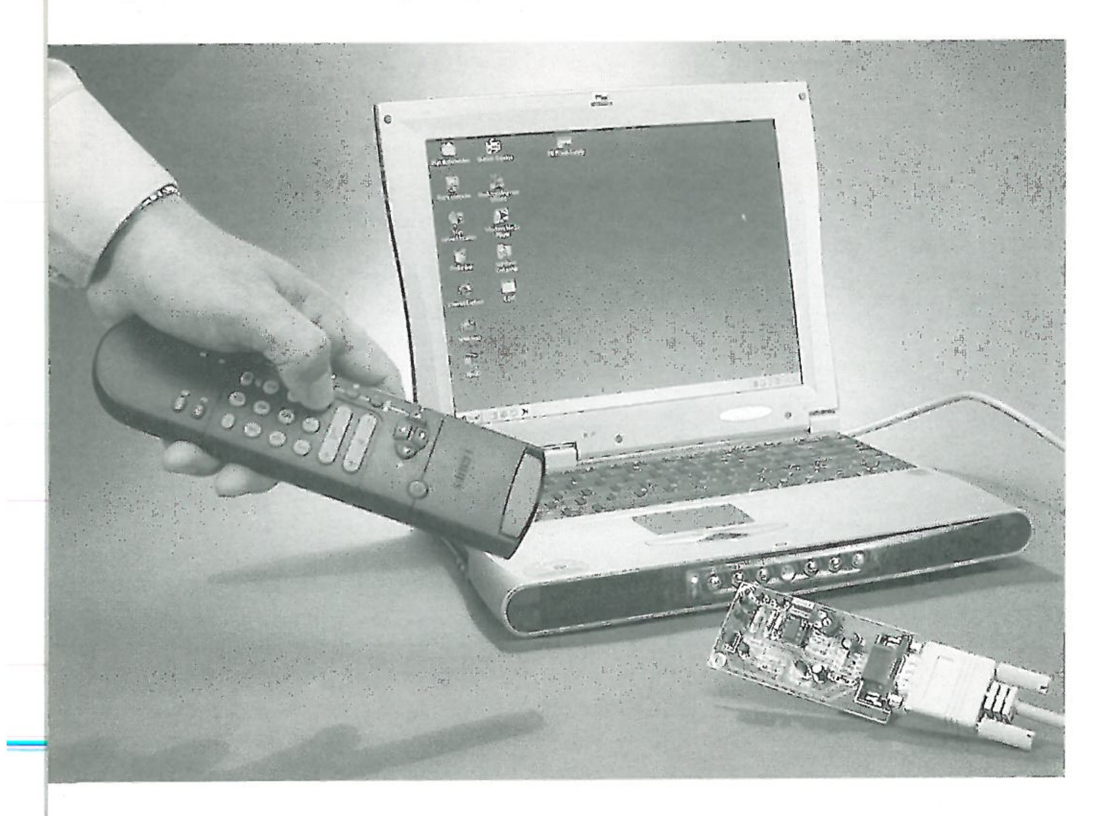
(010204-1)

# Transceptor de infrarrojos para PC (I)

## combina control de datos y enlace de datos

Diseñado por B. Kainka

**Este artículo describe un transceptor que podemos utilizar con nuestro PC para control remoto y transferencia de datos. Se basa en el ya publicado en la serie 'Diseño de periféricos para PC', que también proporciona información sobre los principios de las técnicas de transferencia utilizadas y tecnología de interfaces, las cuales podemos usar para nuestros propios proyectos.**



La transferencia de datos por infrarrojos cada vez es más importante. Los controles remotos de TV utilizan luz infrarroja, pero también se usan en PCs, teclados, impresoras y otros periféricos. La luz infrarroja emplea también cables de fibra óptica. En una aplicación, el transceptor usado aquí representa una continuación de las series de artículos de interfaces para PC y Visual Basic que aparecieron en el curso de 'Diseño de periféricos serie para PC'.

El control remoto por infrarrojos para aparatos de TV, vídeo grabadores y otros dispositivos electrónicos utilizados frecuentemente usa el código estándar de Philips RC5. Es bastante fácil determinar si un control remoto emplea RC5 mediante el programa descrito en este artículo. Este estándar usa señales de luz moduladas a una frecuencia en el rango de 30 a 40 KHz. La unidad de control remoto transmite 'ráfagas', las cuales son pulsos individuales empaquetados. En nuestro caso, esas ráfagas tienen una duración de casi 0,888 ó 1,776 ms. A una frecuencia de modulación de 36 KHz una ráfaga corta tienen 32 pulsos individuales, mientras que una ráfaga larga tiene 64 pulsos. Un paquete completo de datos tiene una duración de aproximadamente unos 25 ms y se repite cada 100 ms cuando se mantiene pulsado un botón.

Una unidad de control remoto por infrarrojos se puede utilizar fácilmente para otros propósitos. Por ejemplo, se puede emplear para controlar ciertas funciones de nuestro propio programa. Una aplicación típica es el control de un PC. Después de comenzar el programa, podemos vol-

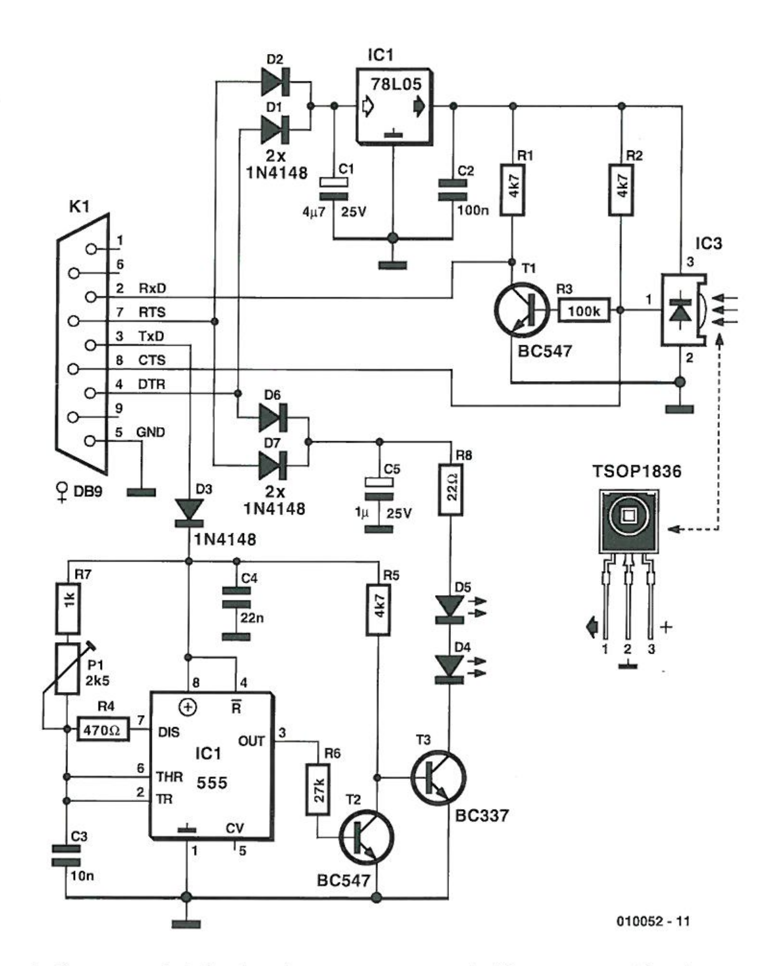



Figura 1. Esquema del circuito de un transceptor de IR para conexión al puerto serie de un PC.

ver y controlar todo desde donde estamos sentados.

## Hardware del transceptor

Gracias a la disponibilidad de receptores integrados, la recepción de señales estándar infrarrojas es relativamente sencilla. El conocido sistema SHF506 de Siemens está disponible con frecuencias de modulación fija de 30, 33, 36 KHz y así sucesivamente. La curva del filtro es relativamente estable, por lo que una desviación relativa de unos pocos kilohercios provocará una reducción de sensibilidad. El Vishai/Telefunken TSOP1836 es un circuito integrado similar. Ambos tipos de circuitos integrados necesitan una tensión de alimentación de –5 V y consumen menos de 2 mA. Se pueden alimentar directamente desde una interface serie del PC.

El transceptor IR descrito aquí tiene un transmisor de IR modulado además del receptor. Esta luz transmitida trabaja con una frecuencia de portadora entre 30 y 40 KHz. También se puede utilizar para el control remoto de dispositivos tales como grabadores de vídeo y aparatos de TV, pudiéndose emplear igualmente para la transferencia de datos entre dos PCs.

El diagrama del esquema se representa en la Figura 1 y muestra el integrado receptor (IC3) y un regulador de tensión (IC1), un 78L05. La tensión de alimentación se toma de las señales de salida DTR y RTS de la propia interface RS-232, las cuales están conectadas a través de los diodos de aislamiento D1 y D2. Puede activarse una tensión de unos 10 V utilizando un programa que se ejecuta en el PC. Esos dos LEDs también transmiten parte de la potencia del circuito a través de D6 y D7.

Debido a que son necesarios pulsos de gran corriente para la transmisión, se usa un condensador electrolítico, C1, de 4,7 µF, para suavizar la tensión de entrada del regulador de tensión.

Si el circuito integrado receptor recibe una señal infrarroja modulada a 36 KHz, produce una señal de salida en su pin medio, que activa un nivel bajo. Esos pulsos de salida se conectan directamente al pin CTS, donde deben ser decodificados mediante software. Aquí es necesaria una resistencia de pull-up, porque la señal CTS tiene una resistencia de entrada relativamente baja. La señal también se invierte por medio de una etapa de transistor (T1) y se aplica a la entrada RxD de la interfaz serie. Este pin sirve para la recepción de datos rápida, por ejemplo desde un enlace de datos entre dos PCs.

El transmisor infrarrojo consta de una etapa de modulación (IC2) y un amplificador de pulso (T2 y T3) que excita a dos diodos infrarrojos (D4 y D5). IC2 es un circuito integrado 555 cableado como oscilador, el cual genera pulsos negativos estrechos con una anchura de unos 2 ms. La frecuencia se puede establecer entre 30 y 40 KHz, utilizando P1. Dependiendo de la aplicación, el potenciómetro puede servir para sintonizar el circuito a la frecuencia adecuada, en lo que se refiere a conseguir el mayor rango posible. El 555 recibe su tensión de alimentación desde el terminal TXD, el cual modula el transmisor poniéndolo en 'ON' o en 'OFF'.

La energía de los dos diodos transmisores de IR también llega desde la interface serie. Las señales DTR y RTS cargan un condensador de 1 µF (C5) a través de D6 y D7. Los breves pulsos de la salida del circuito temporizador (pin 3) fuerzan la etapa de salida que dispone de T2 y T3 en conducción total. Esto da lugar a pulsos de corriente de unos 200 a 300 mA. El rango que se alcanza es de alrededor de 10 m, aunque la corriente de carga de la interface serie es relativamente pequeña a 40 mA, habiendo bastante tiempo entre los pulsos para permitir la recarga del condensador.

Este simple hardware se puede construir usando la placa de circuito impreso mostrada en la Figura 2. Todo lo que necesitamos cuando fijemos los componentes es asegurarnos que los diodos, condensadores electrolíti-

## Transceptor IR

**Especificaciones técnicas**

- Frecuencia de recepción: 30, 33 ó 36 kHz, dependiendo de la versión del circuito integrado.
- Frecuencia de transmisión: 30–40 kHz, ajustable de forma continua.
- Tensión de alimentación: desde la interface serie.
- Rango: approx. 10 m.
- Receptor IR: para control remoto por estándar RC5.
- Transmisor IR: RC5 compatible.
- Transceptor de datos IR: datos serie, 2.400 baudios max.

cos y circuitos integrados están soldados o insertados de forma correcta. Deberíamos evitar el error común de utilizar un conector sub-D hembra en lugar de un conector sub-D macho (9 pines).

## RC5 decodificador software

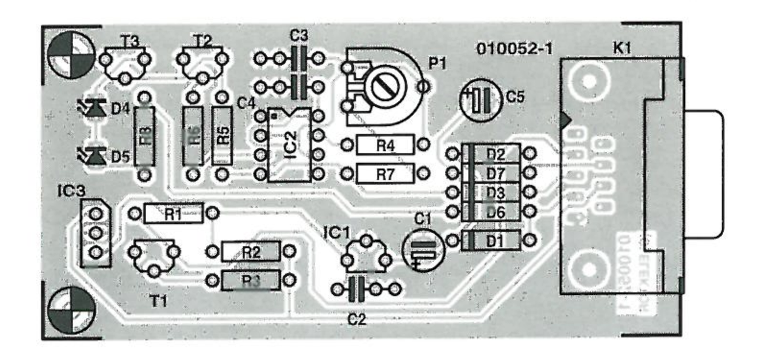
Los datos de salida del integrado receptor están conectados directamente a la señal CTS. Las señales de la unidad de control remoto se han demodulado por el integrado receptor que aparece en este circuito. Un programa para decodificar la señal sólo tiene que evaluar los pulsos entrantes en lo que se refiere a reconocer qué botón se ha pulsado en el control remoto (mando a distancia).

La Figura 3 muestra una señal recibida desde una unidad de control remoto RC5. El diagrama fue capturado utilizando un analizador lógico. Un programa graba directamente los cambios del nivel de la señal en el terminal CTS. El protocolo RC5 utiliza la llamada señal bifase, con la información real contenida en los cambios de fase. Los niveles de señal cambian cada 1,776 ms. El receptor se puede resincronizar de forma continua por medio de estos cambios.

La señal de comienzo con una secuencia de inicio siempre es la misma. Siguiendo esto tenemos tres regiones de datos, cuyos niveles están espaciados 1,776 ms lo cual representan los bits de datos reales. Siguiendo cada cambio de nivel, el receptor primero espera un tiempo no mayor de 0,888 ms e impide que pueda ocurrir cualquier cambio en este intervalo. El siguiente cambio de nivel es una señal de sincronización y un bit de datos. En principio, esta técnica se puede utilizar para transferir palabras de datos de cualquier longitud deseada. En el caso de señales RC5, la longitud de palabra es exactamente de 12 bits, compuesta de la siguiente forma:

- El bit de control (Ctl) cambia entre 0 y 1 cada vez que se pulsa un botón. El receptor puede usar esta información para decidir si se ha pulsado un botón y se ha mantenido sólo una vez o se ha pulsado varias veces de forma sucesiva.
- La dirección del dispositivo (Addr) consta de cinco bits, con el bit más significativo; algunas direcciones de dispositivos estándar son '1' para televisión y '5' para vídeo. La dirección del dispositivo permite usar diferentes controles remotos en la misma sala.
- La región de datos (Dat) consta de seis bits para 64 botones diferentes. Los números de botones (0-9) generan códigos de '0' a '9'. Aquí, el bit más significativo se transmite primero.

El listado 1 muestra el programa decodificador en Visual Basic. La rutina 'RC5' recibe



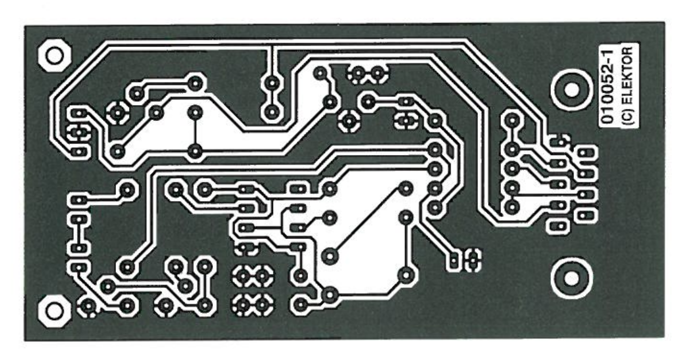




Figura 2. Placa de circuito impreso y componentes para el transceptor de IR (placa no disponible en el Servicio de Lectores).

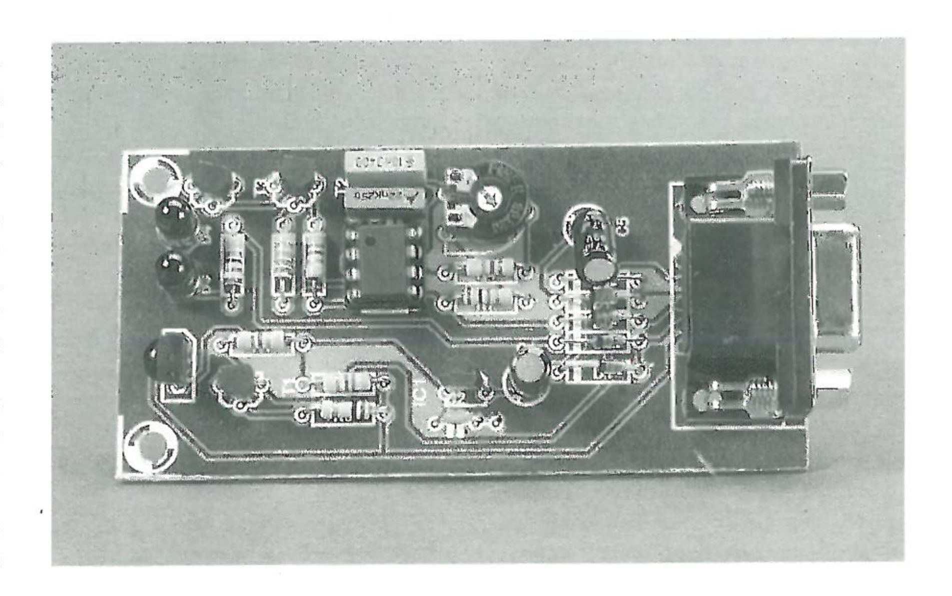
### LISTA DE MATERIALES

**Resistencias:**
R1,R2,R5 = 4k7
R3 = 100kΩ
R4 = 470Ω
R6 = 27kΩ
R7 = 1kΩ
R8 = 22Ω
P1 = 2k5 preset

**Condensadores:**
C1 = 4µF7 25V radial
C2 = 100nF
C3 = 10nF
C4 = 22nF
C5 = 1µF 25V radial

**Semiconductores:**
D1,D2,D3,D6,D7 = 1N4148
D4,D5 = IR-LED, por ejemplo LD271
T1,T2 = BC547
T3 = BC337
IC1 = 78L05
IC2 = 555
IC3 = TSOP1836, SFH506-36

**Varios:**
K1 = conector sub-D 9 hembra, acodado para montaje PCB



## Listado 1

### Recepción y visualización de RC5

```
'
Dim Ctr
Dim Adr
Dim Dat

Sub RC5Error()
  Ctr = -1
  Adr = -1
  Dat = -1
End Sub

Function RC5Bit() As Integer
  TIMEINITUS
  If CTS = 0 Then
    While ((TIMEREADUS < 500) And (CTS = 0))
    Wend
    If TIMEREADUS > 499 Then RC5Error
    DELAYUS 444
    If CTS = 0 Then RC5Error
    RC5Bit = 0
  Else
    While ((TIMEREADUS < 500) And (CTS = 1))
    Wend
    If TIMEREADUS > 499 Then RC5Error
    DELAYUS 444
    If CTS = 1 Then RC5Error
    RC5Bit = 1
  End If
End Function

Sub RC5()
  Ctr = 0
  Adr = 0
  Dat = 0
  Startbit = False
  REALTIME True
  TIMEINIT
  While Not Startbit
    While (CTS = 1) And (TIMEREAD < 500)
    Wend
    Startbit = True
    DELAYUS 444
    If CTS = 1 Then Startbit = False
    DELAYUS 888
    If CTS = 0 Then Startbit = False
    DELAYUS 888
    If CTS = 1 Then Startbit = False
    Adr = Startbit
    If TIMEREAD > 499 Then Startbit = True
  Wend
  DELAYUS 888
  Ctr = RC5Bit
  Adr = 0
  For N = 1 To 5
    DELAYUS 888
    Adr = Adr * 2
    Adr = Adr + RC5Bit
  Next N
  Dat = 0
  For N = 1 To 6
    DELAYUS 888
    Dat = Dat * 2
    Dat = Dat + RC5Bit
      Next N
  REALTIME (False)
End Sub

Private Sub Form_Load()
  OPENCOM "COM2"
  DTR 1
  RTS 1
End Sub

Private Sub Form_Unload(Cancel As Integer)
  CLOSECOM
End Sub

Private Sub Timer1_Timer()
  RC5
  Text1.Text = Str$(Ctr) + " " + Str$(Adr) + " " +
Str$(Dat)
End Sub
```

el data. Aquí la DLL del PORT.DLL del libro 'Interfaces de PC' bajo Windows se utiliza para todos los accesos a la interface serie y para el control de la temporización. La dll PORT.DLL se puede descargar de la página de Elektor Electronics (véase la nota al final del artículo). Esta cuestión es relativamente crítica y requiere el uso de REALTIME=True. Esta rutina espera inicialmente un nivel bajo, el cual actúa como un pulso inicial. En lo que se refiere a evitar que el PC entre en un bucle infinito si no está presente ninguna señal, se construye una condición de timeout. Si no se

## Listado 2

### Carga de imágenes para visualizar en monitor

```
Private Sub Timer1_Timer()
  RC5
  If Dat <> Dat_old Then
   If Dat = 1 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast11.jpg")
   If Dat = 2 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast21.jpg")
   If Dat = 3 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast31.jpg")
   If Dat = 4 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast41.jpg")
   If Dat = 5 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast51.jpg")
   If Dat = 6 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast61.jpg")
   If Dat = 7 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast71.jpg")
   If Dat = 8 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast81.jpg")
   If Dat = 9 Then Picture1.Picture = LoadPicture("D:\Homepage\Bast91.jpg")
  End If
  Dat_old = Dat
End Sub
```

ha recibido ninguna señal después de 500 ms, el programa termina con un mensaje de error.

Los controles por infrarrojos siempre están sujetos a interferencias de otras fuentes de luz. Las lámparas fluorescentes, las cuales generan multitud de picos, son fuentes de interferencia típicas. La rutina de recepción RC5 chequea la secuencia de inicio de la señal recibida para verificar que es correcta. Si la señal sale de una secuencia de pulsos, se puede esperar que tenga un pico de ruido. En este caso, la rutina espera que comience la siguiente secuencia. Esto se hace para asegurar que se reciben señales RC5, incluso en un medio con un alto nivel de interferencias.

Después de la secuencia de inicio se leen los bits individuales mediante una rutina 'RC5bit'. Si leemos un nivel bajo al principio de la rutina debería representar un bit 0. Después, la rutina espera para que un nivel de señal cambie. Después de una anchura de medio pulso (444 ms), se hace un nuevo




Figura 5. Una pequeña ventana en la pantalla del PC.

requerimiento para saber si el mismo nivel está aún presente. Si no es el caso, se detecta un error y todos los datos leídos en este punto se sobreescriben con valor –1. Después de más de 444 ms, la rutina 'RC5bit' devuelve el valor del bit que hemos leído. La rutina de llamada ('RC5') evalúa este bit y espera 444 ms antes de hacer una nueva llamada. De esta manera, cualquier cambio de nivel que se pueda presentar es SKIPPED. Esto es adecuado porque el siguiente cambio válido sólo se espera después de 1,776 ms.

La señal de recepción se controla por medio de un temporizador en un programa en VB. Todos los datos recibidos se ven en una ventana de texto. La Figura 4 muestra la pantalla de salida de los datos recibidos para una unidad de control remoto con la dirección de dispositivo '5' (grabador de vídeo) y el botón '7' pulsado.

Si no se recibe señal se visualiza un mensaje de error adecuado con Ctl= -1 y Addr= -

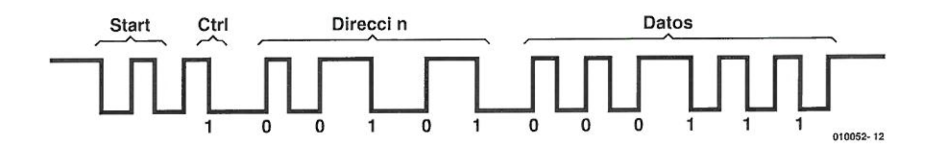



Figura 3. Muestra de señal RC5 con dirección '5' y el botón '7' presionado.

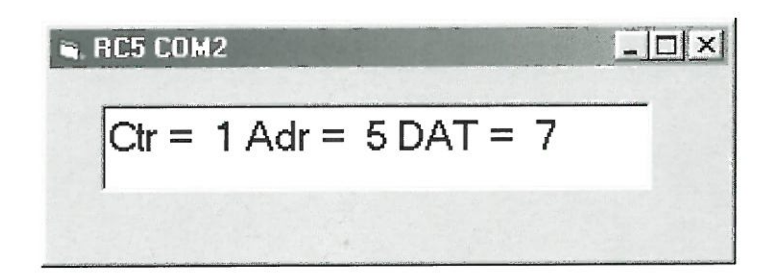



Figura 4. Salida de datos RC5.

1. En una versión operativa de programa, el error de información no debe visualizarse, en su lugar lo utilizaremos para distinguir los datos útiles de los inútiles.

Podemos utilizar este programa para familiarizarnos con los códigos individuales de nuestra propia unidad de control remoto. Esto puede formar la base para posteriores aplicaciones de control, en las cuales sustituiremos una unidad de control remoto. Puede completar la lista de los códigos utilizados comúnmente mediante Internet.

## Control remoto para PC

Otra aplicación útil es el control remoto del PC. Por ejemplo, podríamos controlar cómo utilizar una unidad de control remoto. El listado 2 muestra un segmento de programa con la rutina de tiempos modificada. Aquí se evalúan nueve códigos de botón diferentes para cargar imágenes desde el disco duro. En realidad este programa sólo soporta nueve imágenes, lo cual puede ser un alivio para todos aquellos que han sufrido algo parecido en las ventanas deslizantes, y además no hay nada que pueda evitar que este número se incremente. Por ejemplo, podríamos utilizar los botones '+' y '-' para controlar la secuencia del desplazamiento.

(010052-1)

*El próximo mes, describiremos el software para el transmisor RC5 utilizando un PC y la placa transceptora de IR. Se puede emplear para el control de un vídeo y otros dispositivos mediante un PC y para transferir datos serie entre dos PCs a través de un enlace infrarrojo.*

# Nota de descarga

La dll PORT.DLL utilizada en este artículo, junto con el listado del programa y la placa de circuito impreso, se pueden descargar de forma gratuita de la página de Elektor Electronics en la dirección www.elektor-electronics.co.uk. En esta página podemos encontrar las páginas a descargar pulsando sobre el botón 'Free Downloads'. El mes siguiente publicaremos el artículo en cuestión.
Los ficheros de software se pueden descargar de la página del autor: *http://home.t-online.de/home/B.Kainka*

# Referencias:

B. Kainka, **PC Interfaces under Windows**,
Elektor Electronics (Publishing), Dorchester (ISBN 0 905705 65 3)

# Kit de reconocimiento de habla "voz extrema"

Cuando se trata de productos de reconocimiento del habla, la compañía Sensory (www.sensoryinc.com), con base en California, ha estado en la vanguardia desde hace algún tiempo. Su última oferta es el Kit de reconocimiento del habla "Voice Extreme" ("Voz Extrema").

Este Kit contiene un completo entorno de desarrollo que permite que las aplicaciones puedan programarse utilizando el lenguaje VE-C, un lenguaje de alto nivel muy similar al ANSI-C. Como podríamos esperar, este lenguaje proporciona una gran potencia a la productividad de los programadores, comparado con los entornos de programación basados en el lenguaje ensamblador, lo que permite el desarrollo de aplicaciones para conversación controlada en un entorno de Windows. El código resultante se compila y se vuelca sobre el módulo de memoria flash "Voz Extrema". En este momento, el módulo puede ejecutarse en la placa de desarrollo o desconectarlo de la misma e introducirlo en el sistema destino. Este Kit se distribuye en el Reino Unido por:

Milford Instruments
Leeds. LS25 5AQ
Tel: 01977 683665
www.milinst.com

El kit de desarrollo contiene una tarjeta de desarrollo física muy versátil que, a su vez, integra el módulo de reconocimiento del habla junto con el programa "Integrated Development Environment (IDE; es decir, Entorno de Desarrollo Integrado), que incluye un editor, el programa "Quick Synthesis" y el depurador del código fuente, los cuales permiten que cualquier fichero de sonido con la extensión ".wav", pueda usarse en cualquiera de las aplicaciones "Sensory RSC".

## Una placa con muchas posibilidades

El kit de desarrollo hardware incluye todos los elementos electrónicos, un micrófono, un altavoz y unos conectores para su conexión a equipos externos. También se incluye un cable RS 232 y un adaptador de tensión de red. En la tarjeta física hay cinco pulsadores, uno de ellos para el reset del sistema, otro para el volcado y ejecución de programas y los tres restantes, junto con sus tres diodos LEDs de estado, están libres para or las aplicaciones y programas que desarrollemos.

En la propia placa se dispone de un área para prototipos que nos permite montar los circuitos adicionales de nuestras aplicaciones específicas. Todas las señales de E/S del módulo están disponibles, junto con las líneas de la tensión de alimentación. También se incluye un conector, por si optamos por construir nuestro circuito adicional de forma externa a dicha placa.

El corazón del kit de desarrollo es el módulo "Voice Extreme". Dicho módulo incluye el microcontrolador RSC 300/364, un oscilador de cristal, una memoria ROM con su programa específico, y una memoria Flash RAM de 2 Mbytes para almacenar el programa de nuestras propias aplicaciones. Este módulo se conecta sobre la placa de desarrollo (y, de forma opcional, sobre nuestra placa de aplicación), a través de un conector






tipo "header" de 34 terminales. Para su uso en producción, el módulo "Voz Extrema" está disponible en un paquete "QFD" o en elementos separados. Este paquete también soporta la reprogramación sobre el propio circuito.

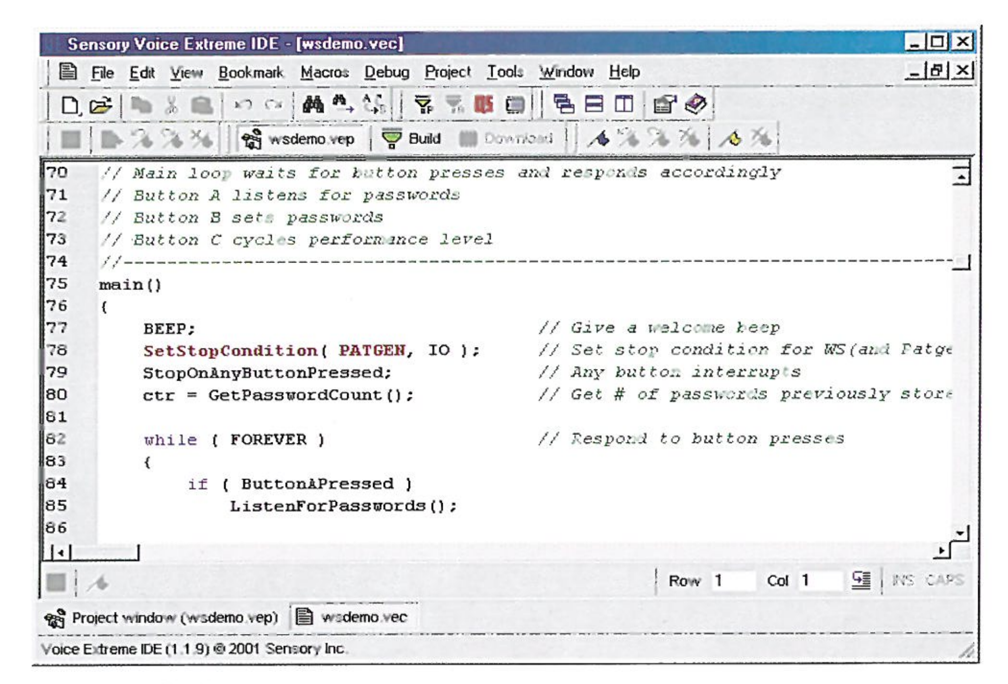
## El programa

El conjunto de programas incluye un Entorno de Desarrollo Integrado (IDE). Este programa estrella contiene todas las herramientas necesarias para escribir el programa de una aplicación específica e incluye un editor C, un compilador, un linkador junto con funciones de ayuda y un cargador de programas. El editor soporta C-Highlightin (C-Mejorado), que mejora las prestaciones del programa.

El IDE se suministra en un CD-ROM junto con el programa "Quick Synthesis" y convierte y comprime ficheros ".wav" en un formato de fichero compatible utilizado por la placa que se incluye. También podemos encontrar en el CD una gran variedad de ejemplos y una extensa documentación.

Ni que decir tiene que para poder escribir los programas de reconocimiento de voz fácilmente es necesario hacer uso de la librería de rutinas que se suministra. Gracias a los numerosos ejemplos incluidos en el CD, junto con la facilidad de ayuda "online", todo se convierte en un sencillo proceso para entender y hacer uso de estas librerías de funciones.

En la documentación de los programas hay que hacer una distinción entre dos tipos de tecnología de reconocimiento del habla: "hablador independiente" y "hablador dependiente". En el primer caso, el programa simplemente reconocerá una señal, pero en el segundo caso reconocerá la señal sólo si proviene de un hablador particular. La diferencia entre las dos tecnologías es una cuestión de la precisión con la que el programa evalúa las señales. Mientras más alta sea la precisión mejor se podrá conseguir especificar la persona que habla y el resultado obtenido. El programa permite que la precisión en el reconocimiento del habla pueda ajustarse en diferentes pasos. Del mismo modo que existen rutinas para el reconocimiento de voz, también hay rutinas para programas de sonido, que pueden usarse para generar música sintética y jugar con ejemplos y fragmentos de sonido.




## En la práctica

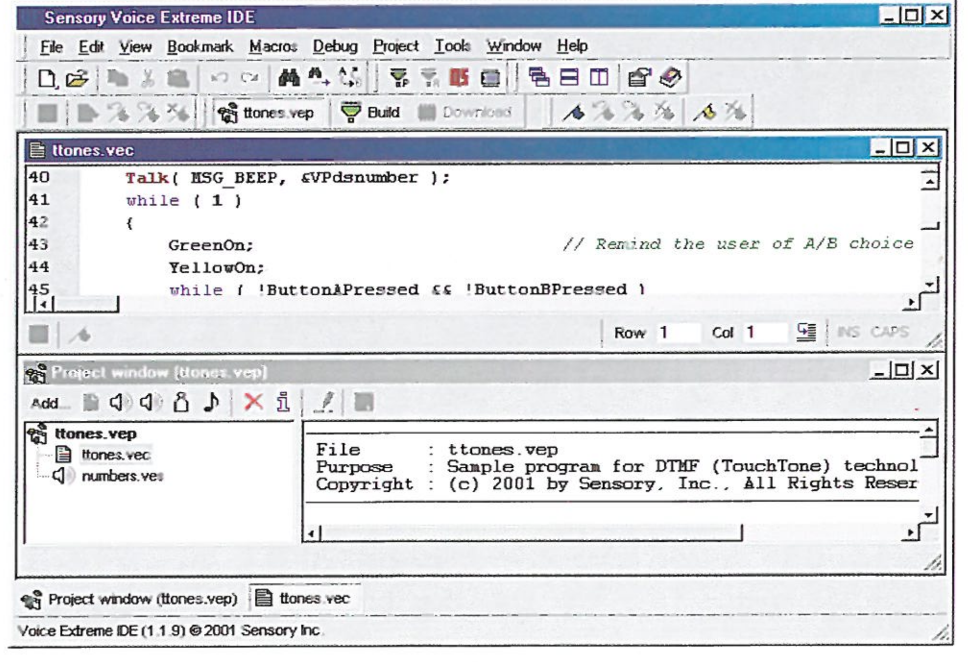
Todos los programas ejemplos incluidos en el CD han sido verificados en el laboratorio de Elektor Electronics. El proceso de reconocimiento del habla funcionó extraordinariamente bien a pesar de la baja calidad del micrófono montado sobre la placa. El fabricante sugiere utilizar un micrófono de alta calidad de sonido para incrementar las prestaciones de reconocimiento de voz del programa.

Los ejemplos solamente demostraron algunas de las posibilidades de esta potente herramienta. Los ficheros en código fuente que lo acompañan nos proporcionaron una buena introducción para saber cómo puede ser utilizada la placa en muy diferentes situaciones, además de suponer un punto de partida ideal para que podamos empezar a escribir nuestras propias aplicaciones.

Además de los ejemplos de reconocimiento del habla también se incluyen unas interesantes demostraciones de generación de música y una compresión de muestras de tonos que trabajan perfectamente con la librería suministrada.

El kit es ideal para todo aquel que esté interesado en el campo de reconocimiento de voz y, especialmente, si ya tenemos una aplicación en mente. Todo lo que necesitamos tener es un ordenador donde esté funcionando Windows 95 ó 98, con 16 Mbytes de memoria RAM, 15 Mbytes de espacio libre en el disco duro, y un puerto serie RS 232 sin usar, junto con una pequeña experiencia de programación en lenguaje C. Por supuesto, también necesitamos el kit de desarrollo, cuyo precio se aproxima a los 170 euros + IVA.

(010085-1)



# Música Industrial Protección de CDs de Audio

## introduciendo errores intencionados para evitar copias

Por Harry Baggen

**Las copias ilegales de CDs de audio cuestan a la industria musical un dineral cada año. Un gran número de compañías usan sofisticadas técnicas para intentar proteger sus CDs contra las copias. Sin embargo, es cuestionable que esto ayude realmente y que el consumo resulte beneficiado.**




En los últimos años, la industria musical se ha complicado con enormes pérdidas producidas por las copias ilegales de CDs de audio. La mayoría de los propietarios de ordenadores tienen algún grabador de CD en su sistema, y los grabadores rápidos sólo tardan unos minutos en hacer una copia de CD para un amigo o vecino. Naturalmente, esto es una suposición, no es un hecho constatable. Con esto no sólo pierde el productor / distribuidor sino también los músicos y compositores que han puesto lo mejor de ellos en cada CD. De acuerdo con la ley, podemos hacer copias de una fotografía grabada o CD para nuestro propio uso, pero no comprar un CD y después hacer diez copias (o que alguien haga diez copias) para todos los amigos.

En lo que se refiere a poner fin a las copias caseras, se han tomado varias medidas sobre métodos de protección de CDs de audio contra copia digital o en ordenador. Varias compañías tienen ahora diferentes sistemas y proclaman que sus métodos funcionan perfectamente. Hay varios productores de música que utilizan estos métodos, pero a menudo sin saber qué CDs están afectados.

Los inventores de varios métodos de protección son reacios a proporcionar mucha información sobre las técnicas utilizadas, pero se basan esencialmente en la modificación de datos en el CD, de tal forma que un PC no pueda hacer una copia exacta del CD utilizando una unidad de CD-ROM. La técnica más conocida es la utilizada por Macrovisión. En su sistema 'Seguro de audio', los datos de audio y códigos de corrección de error en el CD son intencionadamente mutilados en el punto en el que una unidad de CD-ROM en un ordenador tendrá problemas de manejo de datos. Como resultado de esto, un fichero leído digitalmente producirá

# elektor

# PEDIDO DE NÚMEROS ATRASADOS




Sí, deseo recibir contra reembolso los números de **ELEKTOR** que indico a continuación, al precio de portada, más gastos de envío. (Se servirán sólo los números que no estén agotados al tiempo de recibir la orden de pedido).

**Deseo Recibir los números:** ______________________________

Nombre:.................................. Apellidos:..............................................................

Dirección:.................................................................. N°:........ Piso:.......... Tel.:.................................

C.P.:............. Población:.................................................................. Provincia:.............................

**ENVIAR ESTE CUPÓN A LARPRESS, S.A. C/ LA FORJA, 27-29 POL. IND. TORREJÓN DE ARDOZ 28850 MADRID. TEL.: 91 677 70 75 FAX: 91 676 76 65**

Usted tiene derecho a acceder a la información que le concierne, recopilada en nuestro fichero de datos y a cancelarla o rectificarla de ser erronea. Si no desea recibir información comercial de esta u otras empresas, le rogamos nos lo haga saber mediante comunicación escrita con todos sus datos personales.

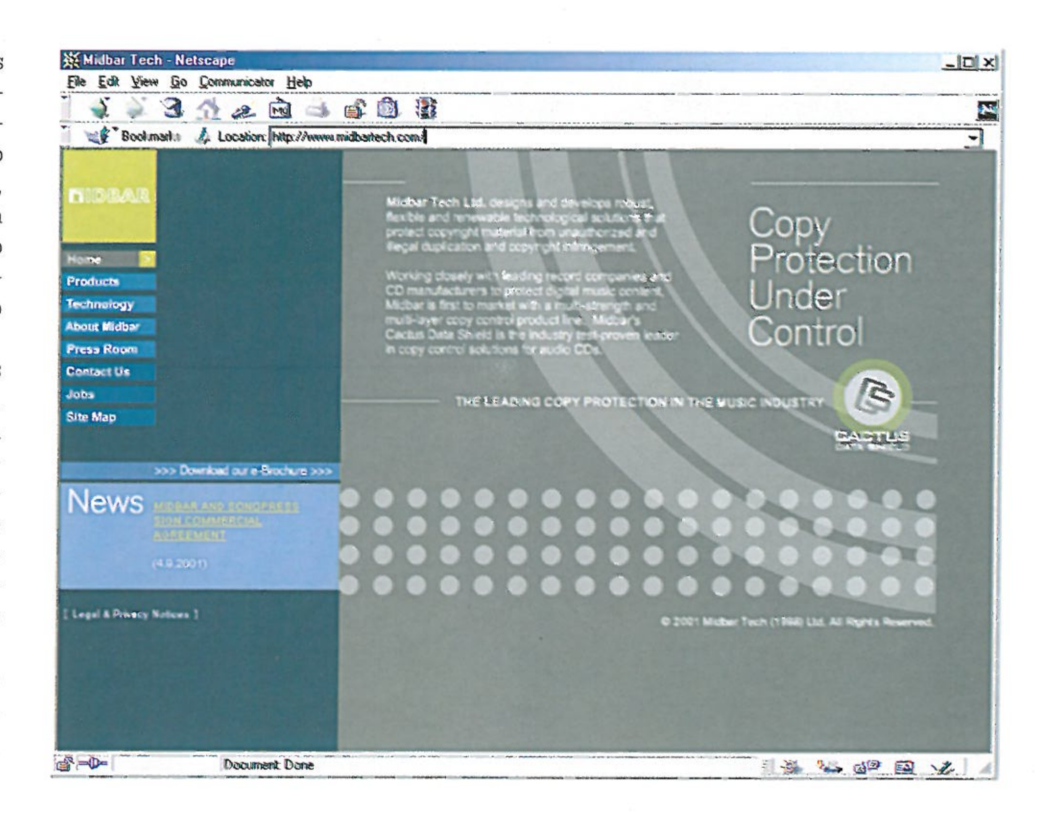
toda clase de ruidos e interferencias en la reproducción, por lo tanto no será fácil copiar un CD-R. Los reproductores de audio CD normales no tienen problemas con tales datos, porque incluso teniendo el sistema de corrección de errores y leyendo datos confundidos, el reproductor inventa valores intermedios por medio de interpolación.

Nosotros no encontramos tales métodos particularmente atractivos. En primer lugar, la industria hace un CD estándar que proporciona la reproducción de calidad más pura posible con un alto grado de errores de tolerancia (para compensar posibles ruidos), de forma que sólo se perciban errores grandes, introducidos de forma intencionada.

Las protestas contra este método están llegando lentamente pero son cada vez más y más numerosas, particularmente por parte de los audiofilos, quienes temen que las técnicas de protección puedan tener un efecto negativo sobre la calidad del sonido (y nosotros debemos decir que estamos completamente de acuerdo). Entre otros, una de las personas más relevantes a nivel mundial en el tema del hi-fi, el inglés Martin Colloms, es absolutamente contrario a tales métodos. Él compara esto con borrar los cuadros colgados en las galerías de arte para evitar que sean robados. Podemos encontrar más artículos sobre este tema en las publicaciones New Scientist [1] y New Media Music [2]. En los Estados Unidos ya se ha presentado una demanda contra una compañía de música por protección en los CDs sin ponerlo explícitamente en el envoltorio.

¿Qué compañías están involucradas en la protección de CDs? La más conocida y grande es la ya mencionada Macrovisión [3], conocida entre otras cosas por el sistema de protección de vídeo del mismo nombre.

La técnica de'Audio seguro' de dicha compañía fue desarrollada por la compañía Israelé TTR Technologies. Macrovision dijo que llevaban realizando pruebas durante varios meses utilizando personas expertas que intentaban detectar cualquier cambio en las señales de audio de las cintas copiadas, sin conseguirlo. Sin embargo, ellos no identificaban los CDs en los que se había aplicado esta técnica, aunque hay un importante número de títulos que se han producido en un gran número y que tienen un mínimo de complicaciones (de acuerdo a Macrovision).




Sony, además de hacer productos de consumo, también es uno de los mayores productores de la industria del audio y de CD (Sony DAC C). Utiliza una técnica propia llamada Key2Audio [4], que aseguran que no corrompe los datos musicales, sino que en su lugar se añade una especie de huella digital cuando se hace el cristal master del CD, con el resultado de que la copia digital no es lo más larga posible.

Cactus Data Shield es una técnica desarrollada por Midbar [5] y utilizada ampliamente. Aparentemente, al menos un millón de CDs ya han sido protegidos utilizando este sistema. No se sabe mucho sobre él.

La técnica MediaCloQ de SunnComm [6] emplea otro método. Un CD que se ha protegido utilizando este método se puede reconocer por una reflexión claramente diferente desde la región situada al final de la pista de audio del CD.

La página web de CD Media World [7] proporciona un buen sumario de todos los sistemas de protección, con una breve explicación de cada sistema.

Con todo lo que acabamos de decir sobre la protección de copias de CDs de audio, no deberíamos pensar que es muy fácil copiar un CD mediante una salida de audio normal. La principal desventaja de esto es que en poco tiempo se producirá una importante pérdida en la calidad del sonido debido a las conversiones D/A y A/D. La cuestión es si el consumidor está realmente concienciado (después de todo, la mayoría están muy contentos de poder descargar ficheros MP3 directamente desde Internet).

(025005-1)

## Direcciones de Internet:

[1] NewScientist:
www.newscientist.com/news/news.jsp?id=ns9999998

[2] New Media Music:
www.newmediamusic.com/articles/NM01080292.html
www.newmediamusic.com/articles/NM01100072.html

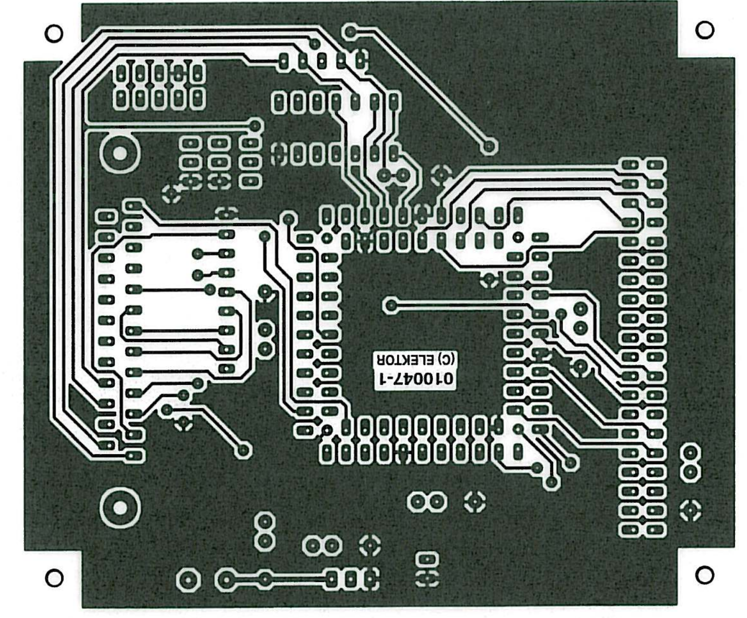
[3] Macrovision SafeAudio:
www.macrovision.com/solutions/newtech/safeaudio.php3

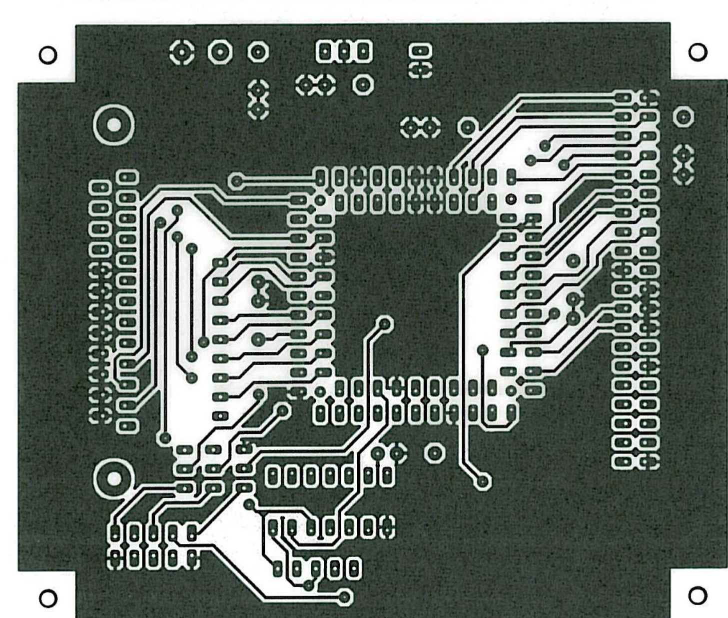
[4] Sony DADC Key2Audio:
www.key2audio.com/key2audio/index.htm

[5] Midbar Cactus Data Shield
www.midbartech.com/cactus2.html

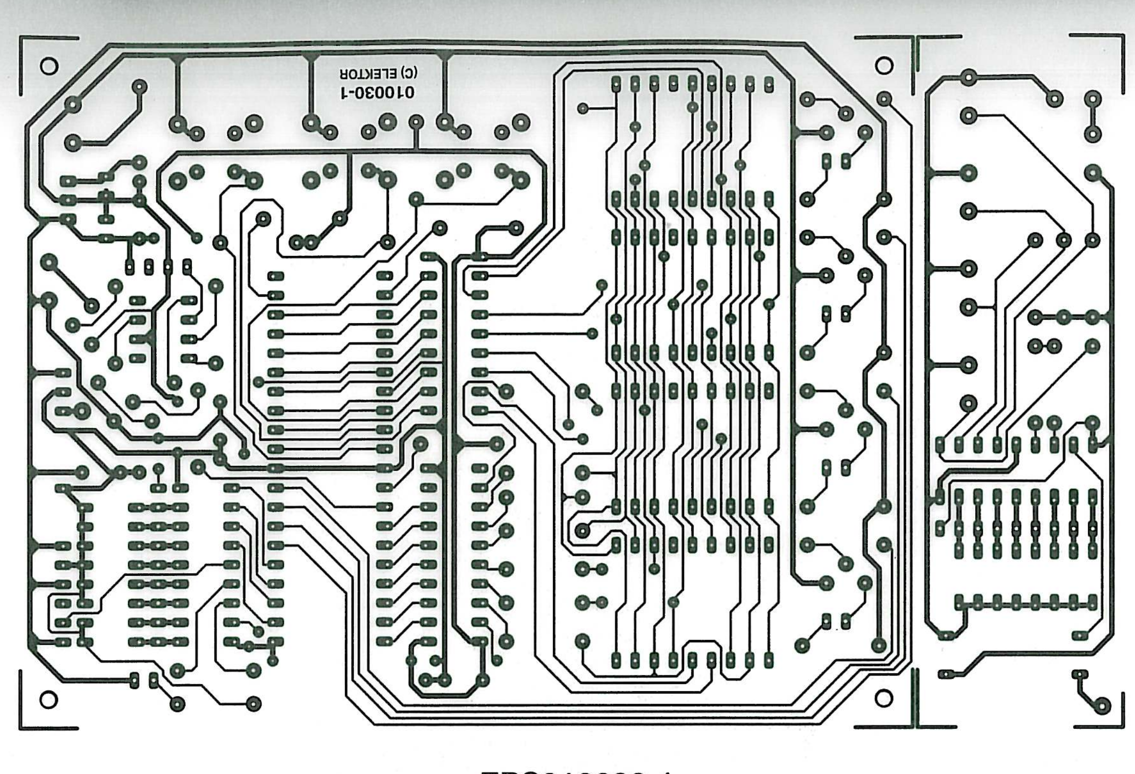
[6] SunnComm MediaCloq:
www.sunncomm.com/

[7] CD Media World:
www.cdmediaworld.com/hardware/cdrom/cd_protections.shtml

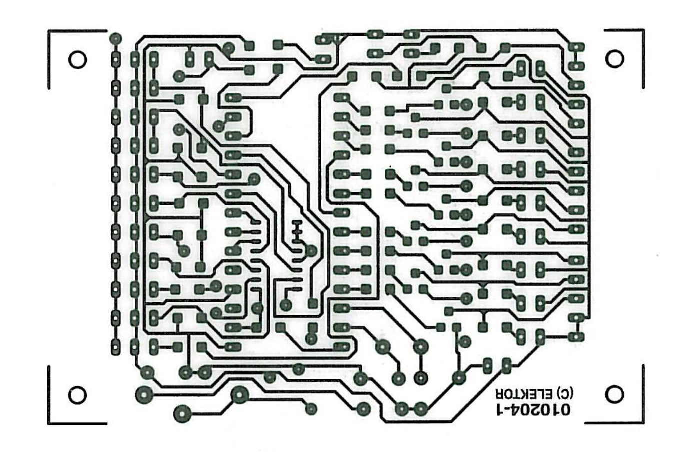


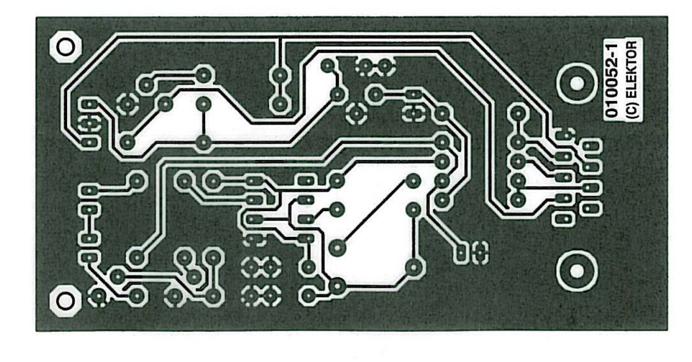
EPS010047-1
Interfaz de disco duro para puerto de impresora.
Circuito impreso de doble cara.




EPS010030-1
Interrogador maestro.




EPS010204-1
Iluminación y caja de cambios.




EPS010052-1
Iluminación y caja de cambios.